3.2.4.2.6 Adding a Task from a Failed Build in Eclipse ................................................. 127
3.2.5 Using FishEye in the Eclipse Connector ............................................................ 129
  3.2.5.1 Configuring your FishEye Options in Eclipse ............................................. 129
  3.2.5.2 More about Configuring FishEye Repositories in Eclipse ......................... 132
  3.2.5.3 Working with your FishEye Repository View in Eclipse ............................. 133
3.2.6 Using Crucible in the Eclipse Connector .......................................................... 134
  3.2.6.1 Configuring your Crucible Options in Eclipse ........................................... 135
  3.2.6.2 Working with Crucible Reviews in Eclipse ................................................ 137
    3.2.6.2.1 Monitoring Review Updates in Eclipse .............................................. 142
    3.2.6.2.2 Viewing a Review in Eclipse .......................................................... 143
    3.2.6.2.3 Managing Review Comments in Eclipse ............................................ 145
    3.2.6.2.4 Working on a Review in the Diff View in Eclipse ............................... 149
    3.2.6.2.5 Creating a Post-Commit Review from within Eclipse ............................ 152
    3.2.6.2.6 Creating a Patch Review from within Eclipse .................................... 155
    3.2.6.2.7 Conducting a Pre-Commit Review from within Eclipse ......................... 159
    3.2.6.2.8 Assigning Reviewers in Eclipse ..................................................... 165
    3.2.6.2.9 Adding Changesets or Patches to Review in Eclipse ............................ 166
    3.2.6.2.10 Using the Crucible Review Perspective ........................................... 171
3.2.7 Eclipse Connector FAQ ....................................................................................... 174
  3.2.7.1 Configuration FAQ for the Eclipse Connector ............................................. 174
    3.2.7.1.1 Using NTLM Authentication with the Eclipse Connector ......................... 174
    3.2.7.2 Usage FAQ for the Eclipse Connector ..................................................... 174
      3.2.7.2.1 Custom Fields Do Not Work As Expected ........................................ 175
      3.2.7.2.2 Customised Priorities Do Not Work ............................................. 175
      3.2.7.2.3 Error Parsing Server Response - No Project Found ........................... 175
      3.2.7.2.4 Estimated Time Displayed in the Task Editor Differs from the Web Interface 175
      3.2.7.2.5 Invalid Date Error When Submitting a JIRA Issue ............................... 176
      3.2.7.2.6 Issue Dependencies Do Not Display in Task List ............................. 176
      3.2.7.2.7 Missing Attachments in Task Editor ............................................. 176
      3.2.7.2.8 Unexpected Redirect Errors when Submitting an Issue .................... 176
  3.2.7.3 Installation FAQ for the Eclipse Connector ............................................... 177
    3.2.7.3.1 Error while Collecting Items to be Installed .................................... 177
    3.2.7.3.2 Installation Error - Cannot Find a Solution Satisfying the Following Requirements 177
3.2.8 Eclipse Connector Resources ......................................................................... 179
  3.3 Atlassian IDE Connector Release Notes ............................................................. 180
  4. TreeNavigation ..................................................................................................... 180
The children of this page contain information which is included in other pages. This is a library of re-usable information chunks.

If you want to change any of these pages, be aware that:

- Changing page names is problematic — you will need to change all the include and excerpt-include macros manually.
- The content is used in many places — make sure your change is generic enough to fit the contexts in which the pages are used.
- It is possible that the entire content of a page is included into other pages and other spaces. This is particularly true if the page does not contain [excerpt] tags. When adding content, make sure it is OK for including into other pages. Don’t add a “Related Topics” section or high-level headings.

__newreleaseIDEPlugin__

Current released version:

Version 2.2.3 of the Atlassian Connector for IntelliJ IDEA has now been released — see the Atlassian Connector for IntelliJ IDEA - v2.2.3 Release Notes.

__newreleaseIDEPluginForEclipse__

Current released version:

Version 2.0 Beta of the Atlassian Connector for Eclipse has now been released — see the release notes.
Current released version:
Version 1.0 Beta of the Atlassian Connector for Visual Studio has now been released — see the Atlassian Connector for Visual Studio - 1.0 Beta Release Notes.

About the IDE Plugin for Eclipse

The Atlassian Connector for Eclipse is an Eclipse plugin. It allows you to work with the Atlassian products within your Eclipse IDE. Now you don't have to switch between websites, email messages and news feeds to see what's happening to your project and your code. Instead, you can see the relevant JIRA issues, Crucible reviews and Bamboo build information right there in your development environment.

About the IDE Plugin for IDEA

The Atlassian Connector for IntelliJ IDEA is an IntelliJ IDEA plugin. It allows you to work with the Atlassian products within your IDE. Now you don't have to switch between websites, email messages and news feeds to see what's happening to your project and your code. Instead, you can see the relevant JIRA issues, Crucible reviews and Bamboo build information right there in your development environment.

Configuring a Bamboo Server in Eclipse

1. On the Eclipse 'Add Task Repository' screen, select the 'Bamboo' task repository type and click 'Next'.
2. The 'Bamboo Repository Settings' screen appears, as shown below:
3. Enter the following information:
   - **Server** — The location (URL) of your Bamboo server.
   - **Label** — A descriptive name for your Bamboo server, e.g., 'Bamboo Extranet'.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your Bamboo server.
   - **Password** — Your password on the Bamboo server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.
   - If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.
   - Click the 'Refresh' button to verify the information you have entered. The connector will attempt to connect to the Bamboo server. It will also retrieve the latest list of build plans from the Bamboo server.
4. Select the plans that you want the connector to watch. You can either select the plans individually or you can click the 'Favourites' button to choose your favourite plans as defined on the Bamboo server.
5. Click 'Finish' to save the changes.
6. The new task repository appears in your Eclipse 'Task Repositories' view.
7. If necessary, you can now you can adjust the Bamboo settings.
   - Select 'Window', 'Preferences' then open the Bamboo settings panel as shown below:
Enter the 'Auto Refresh Interval Rate'. This is the number of minutes that the connector will wait between calls to fetch new information from your Bamboo server, when the connector is set to refresh automatically.

Tick or untick the 'Refresh Automatically' checkbox. Untick this option if you do not want the connector to poll the server automatically. In that case, you will need to refresh your Bamboo information manually by synchronising the Bamboo task repository.

9. Next step: To see your Bamboo builds, open the 'Bamboo' view. (Select 'Window', 'Show View', 'Other', then open the 'Atlassian' folder.)

You can configure one or more Bamboo repositories, i.e. you can connect to more than one Bamboo server.

Bamboo builds are shown in a Bamboo-specific view
Bamboo has its own 'Bamboo' view. This is different from JIRA and Crucible, where issues and reviews are included into your 'Task List' view. Next, take a look at the Bamboo view.

**Configuring a Bamboo Server in IDEA**

Configuring your Bamboo Server Connections

To configure your Bamboo server connection(s):

1. Go to the 'Project Settings' for the 'Atlassian Connector', by doing one of the following:
   - Open the IDEA 'Settings' dialogue, then go to the 'Project Settings' section and click the 'Atlassian Connector' icon.
   - Or you can click the configuration icon on your connector window.
2. Click the 'Servers' tab.

To add a Bamboo server:

1. Click the plus icon on the configuration panel.
2. A list of server types will appear. Select 'Add Bamboo Server'.
3. A form will appear. Enter the information as follows:
   - 'Server Enabled' — Leave this checkbox ticked (default). If necessary, you can remove the tick to disable particular servers without deleting them. This is useful if your servers are behind a firewall and you don’t have access to them.
   - 'Server Name' — A description of your Bamboo server.
   - 'Server URL' — The address of your Bamboo server.
   - 'Username' and 'Password' — The login name and password you use to access your Bamboo server.
   - Remember Password — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.
   - If you choose to remember the password, it is stored in a Base64 encoding, so it is not really secure.
   - 'Use Default Credentials' — Put a tick in the checkbox if you want to use the single username and password that you have defined as your default credentials. You can set the default credentials on the 'Defaults' tab.
4. Click the 'Test Connection' button to check that the connection to the server works. A list of build plans will appear.
5. If you have a Bamboo build server located in a different time zone than you, you can manually adjust the 'Time Zone Difference'. You should specify a positive difference if your time is ahead of your build server (e.g., you are in Russia and the build server is in the UK). You should specify a negative difference if your time is behind your build server (e.g., you are in the US and your build server is in Spain).
6. Now select the Build Plans that the connector will watch. You can either select plans manually from the list of plans defined on the Bamboo server, or simply use your favourite plans as defined on the server. Your favourite plans are marked with a yellow star.
7. Click 'Apply' to save your changes and continue with server configuration, or 'OK' to save your changes and close the configuration tab.
8. Now you can configure the Bamboo options, as described below.

You can add more than one Bamboo server.

Configuring your Bamboo Options

1. Open the IDEA ‘Settings’ dialogue, then go to the ‘IDE Settings’ section and click the ‘Atlassian Connector’ icon.
2. Define the behaviour of the popup window that is shown when the status of the build changes. (See Working with Bamboo Builds in IDEA.)
3. Set the polling interval that the connector will use to monitor build plans on all defined Bamboo servers. Specify the value in minutes.

RELATED TOPICS
Working with Bamboo Builds in IDEA

Configuring a Crucible Server in Eclipse

- Single server for both Crucible and FishEye = single task repository
  If you have a single server for both FishEye and Crucible, you should set up the Crucible task repository and then select the ‘Crucible Server Contains FishEye Instance’ option as described below. Due to Mylyn requirements, you cannot set up two separate task repositories for the same URL.

1. On the Eclipse ‘Add Task Repository’ screen, select the ‘Crucible’ task repository type and click ‘Next’.
2. The ‘Crucible Repository Settings’ screen appears, as shown below:

   ![Crucible Repository Settings](image)

   - **Server**: https://studio.atlassian.com/source
   - **Label**: STAC Crucible
   - **User ID**: myname
   - **Password**: ********

3. Enter the following information:
   - **Server** — The location (URL) of your Crucible server.
   - **Label** — A descriptive name for your Crucible server, e.g. ‘Extranet Crucible’.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if
your servers are behind a firewall and you do not have access to them at a particular point in time.

- **User ID** — The username you use to connect to your Crucible server.
- **Password** — Your password on the Crucible server, matching the above username.
- **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

   If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. Click the 'Validate Settings' button to verify the information you have entered.
5. Choose a 'Review Activation' setting. These settings determine what happens when you open a source code file from within the Crucible review editor in Eclipse. For example, you may open a Crucible review within Eclipse. The review will open in an Eclipse editor view, and its 'Review Files' section will show the files in the review. You can then click a file to open it in an Eclipse editor. The options are as follows:
   - 'Always' — The Crucible review will be automatically activated if you open a file from the Crucible review editor.
   - 'Never' — The Crucible review will never be automatically activated.
   - 'Prompt' (default) — When you open a file from the Crucible review editor, the connector will ask you whether you want to activate the Crucible review. This is the default setting.

6. If your Crucible server is linked to a FishEye server:
   - Put a tick in the checkbox labelled 'Crucible Server Contains FishEye Instance'.
   - Set up your FishEye mappings, as described in the FishEye section of this documentation.

7. Click 'Finish' to save the changes.
8. Eclipse Mylyn will prompt you to add a new query for the new Crucible repository. This is where you will choose your Crucible filter, to determine which reviews appear in your task list. You can do this now or skip this step and do it later. You can also add more queries later.

You can configure one or more Crucible repositories, i.e. you can connect to more than one Crucible server.

### Configuring a Crucible Server in IDEA

#### Configuring your Crucible Server Connections

To configure your Crucible server connection(s):

1. Go to the 'Project Settings' for the 'Atlassian Connector', by doing one of the following:
   - Open the IDEA 'Settings' dialogue, then go to the 'Project Settings' section and click the 'Atlassian Connector' icon.
   - Or you can click the configuration icon on your connector window.
2. Click the 'Servers' tab.

To add a Crucible server:

1. Click the plus icon on the configuration panel.
2. A list of server types will appear. Select 'Add Crucible Server'.
3. A form will appear. Enter the information as follows:
   - 'Server Enabled' — Leave this checkbox ticked (default). If necessary, you can remove the tick to disable particular servers without deleting them. This is useful if your servers are behind a firewall and you don't have access to them.
   - 'Server Name' — A description of your Crucible server.
   - 'Server URL' — The address of your Crucible server.
   - 'Username' and 'Password' — The login name and password you use to access the Crucible server.
   - 'Remember Password' — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

   If you choose to remember the password, it is stored in a Base64 encoding, so it is not really secure.

   - 'Use Default Credentials' — Put a tick in the checkbox if you want to use the single username and password that you have defined as your default credentials. You can set the default credentials on the 'Defaults' tab.

4. Click the 'Test Connection' button to check that the connection to the server works.
5. If your Crucible server is linked to a FishEye server, put a tick in the checkbox labelled 'Crucible Server Contains FishEye Instance'. Remember to set up your FishEye defaults, as described in Configuring your FishEye Options in IDEA.

   Don't worry if you do not have a FishEye server. There is very little effect on the connector's functionality. The only thing you will not be able to do, is to access a FishEye diff view of the source code under review.

6. Click 'Apply' to save your changes and continue with server configuration.
7. Click the 'Defaults' tab to set up a default Crucible server, project and repository. These defaults will be used when you create a review directly from your source within IDEA.
8. Now you can configure the Crucible options, as described below.

You can add more than one Crucible server.

#### Configuring your Crucible Options

1. Open the IDEA 'Settings' dialogue, then go to the 'IDE Settings' section and click the 'Atlassian Connector' icon.
2. Define the settings as follows:
   - 'Popups' — Define the behaviour of the popup window that is shown when someone adds a Crucible review that affects you. (For an example of the popup itself, see Working with Crucible Reviews in IDEA.) You can choose to see a popup
window whenever a review is added to the Crucible server, or you can choose not to see any popups at all.

- **Background refresh every xx minutes** — Set the polling interval that the connector will use to monitor all defined Crucible servers. Specify the value in minutes. The default is 10 minutes.
- **Timeout review creation after xx minutes** — Specify the number of minutes to allow, after you have initiated a new review, before the review creation will be timed out. The default is 5 minutes.

1. Crucible (or FishEye to be more exact) does not necessarily notice a new Subversion commit immediately. Instead, it polls the repository for changes every minute or so. If you commit some files and then try to create a review right away, this will fail. The connector detects the error and continues attempting to create the review every 10 seconds until it is successful or until it times out.

3. Click 'OK' to save your changes.

**RELATED TOPICS**

Working with Crucible Reviews in IDEA

---

**Configuring a FishEye Server in Eclipse**

If you do not have a Crucible server, you can hook up your FishEye server as an Eclipse task repository without the additional Crucible functionality.

1. On the Eclipse 'Add Task Repository' screen, select the 'FishEye' task repository type and click 'Next'.
2. The 'FishEye Repository Settings' screen appears, as shown below:

![FishEye Repository Settings](image)

3. Enter the following information:
   - **Server** — The location (URL) of your FishEye server.
   - **Label** — A descriptive name for your FishEye server, e.g. 'FishEye Atlassian Developer'.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your FishEye server.
   - **Password** — Your password on the FishEye server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

   If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an
4. Click the 'Validate Settings' button to verify the information you have entered.
5. Click 'Finish' to save the changes.
6. The new task repository appears in your Eclipse 'Task Repositories' view.
7. Now you need to map your source code to your FishEye repository and FishEye project.
   • Select 'Window', 'Preferences' then open the 'Repository Mappings' panel as shown below:

   ![Repository Mappings Panel]

   • Click the 'Add' button to add a new mapping. The 'Add Mapping' screen appears, as shown below:

   ![Add Mapping Screen]

   • Enter the 'SCM Path' — Supply the path to your source code as used locally by the project.
   • Select the FishEye/Crucible 'Server' as configured in your task repositories. (See instructions above.)
   • The connector will retrieve a list of FishEye repositories from the server you have supplied. Select the appropriate 'Source Repository' to map your local project to the FishEye repository.

Some notes about the FishEye integration:

• You can configure one or more FishEye repositories, i.e. you can connect to more than one FishEye server.
• We have chosen to configure FishEye servers as standard Mylyn task repositories. They are not really regular repositories like JIRA or Crucible, because there are no tasks associated with a FishEye task repository. We are using the task repository concept to take advantage of various infrastructural facilities available in Mylyn and to make user experience more consistent with the rest of the Atlassian products. See More about Configuring FishEye Repositories in Eclipse.
Configuring a FishEye Server in IDEA

Configuring FishEye and Crucible Connections on Same Server

If you have FishEye and Crucible running on the same server, then you can use your Crucible server configuration for the connector's FishEye functionality too:

1. Simply put a tick in the checkbox labelled 'Crucible Server Contains FishEye Instance' on the Crucible server configuration screen.
2. Then set up your FishEye defaults, as described below.

Crucible server configuration is described in full in Configuring your Crucible Options in IDEA.

You can add more than one Crucible and/or FishEye server.

Configuring a Separate FishEye Server Connection

To configure a FishEye server connection independently of Crucible:

1. Go to the 'Project Settings' for the 'Atlassian Connector', by doing one of the following:
   - Open the IDEA 'Settings' dialogue, then go to the 'Project Settings' section and click the 'Atlassian Connector' icon.
   - Or you can click the configuration icon on your connector window.
2. Go to the 'Servers' tab.
3. Click the plus icon on the configuration panel.
4. A list of server types will appear. Select 'Add FishEye Server'.
5. A form will appear. Enter the information as follows:
   - 'Server Enabled' — Leave this checkbox ticked (default). If necessary, you can remove the tick to disable particular servers without deleting them. This is useful if your servers are behind a firewall and you don't have access to them.
   - 'Server Name' — A description of your FishEye server.
   - 'Server URL' — The address of your FishEye server.
   - 'Username' and 'Password' — The login name and password you use to access the FishEye server.
   - Remember Password — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.
   - If you choose to remember the password, it is stored in a Base64 encoding, so it is not really secure.
   - 'Use Default Credentials' — Put a tick in the checkbox if you want to use the single username and password that you have defined as your default credentials. You can set the default credentials on the 'Defaults' tab.
6. Click the 'Test Connection' button to check that the connection to the server works.
7. Click 'Apply' to save your changes and continue with server configuration.
8. Click the 'Defaults' tab and set up your FishEye defaults. It is important to configure your FishEye defaults correctly. These values will be used to construct the path when you attempt to open a source file from IDEA in FishEye's web interface. See Working with your FishEye Repository View in IDEA. Set the default values as follows:
   - Server — Select one of your FishEye connections.
   - Repository — Select the repository where your source files reside. These are the source files you will be working on most often in IDEA.
   - Path to Project — Enter the path to the root of the project in your repository. For example, the path may be one of the following:
     - blank, if your project is at the root of the repository.
     - trunk/
     - trunk/myproject

Example of Default Repository and Path

Let's assume:

- Your FishEye instance is located at:
  
  http://example.com/fisheye

- You have a repository called my_project.
- You are working under the trunk directory under this project.

Your settings will be:

<table>
<thead>
<tr>
<th>Default Repository</th>
<th>Path to Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>my_project</td>
<td>trunk/</td>
</tr>
</tbody>
</table>

RELATED TOPICS

Working with your FishEye Repository View in IDEA
Configuring a JIRA Server in Eclipse

1. On the Eclipse 'Add Task Repository' screen, select the 'JIRA' task repository type and click 'Next'.
2. The 'JIRA Repository Settings' screen appears, as shown below:

   - **Server**: https://extranet.atlassian.com/jira
   - **Label**: Extranet JIRA
   - **User ID**: myname
   - **Password**: ****

   - **Server** — The location (URL) of your JIRA server.
   - **Label** — A descriptive name for your JIRA server, e.g. 'JAC'.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your JIRA server.
   - **Password** — Your password on the JIRA server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

3. Enter the following information:
   - **Server** — The location (URL) of your JIRA server.
   - **Label** — A descriptive name for your JIRA server, e.g. 'JAC'.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your JIRA server.
   - **Password** — Your password on the JIRA server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.
If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. If necessary, you can change the additional settings. If in doubt, leave these settings as set by default.
   - The 'Task Editor Settings' section is used to configure the markup language used by the task editor when Mylyn's WikiText extension is installed. By default, the Atlassian Connector for Eclipse will set the markup language to 'Confluence', so that WikiText recognises Confluence wiki markup as used by JIRA. See the Mylyn WikiText User Guide for more information.

5. Click 'Finish' to save the changes.
6. Eclipse Mylyn will prompt you to add a new query for the new JIRA repository. This is where you will choose your JIRA filter, to determine which issues appear in your task list. You can do this now or skip this step and do it later. You can also add more queries later.
7. The new task repository appears in your Eclipse 'Task Repositories' view. Your JIRA issues will appear in your 'Task List' view.

You can configure one or more JIRA repositories, i.e. you can connect to more than one JIRA server.

_Configuring a JIRA Server in IDEA

Configuring your JIRA Server Connections

To configure your JIRA server connection(s):

1. Go to the 'Project Settings' for the 'Atlassian Connector', by doing one of the following:
   - Open the IDEA 'Settings' dialogue, then go to the 'Project Settings' section and click the 'Atlassian Connector' icon.
   - Or you can click the configuration icon on your connector window.
2. Go to the 'Servers' tab.

To add a JIRA server:

1. Click the plus icon on the configuration panel.
2. A list of server types will appear. Select 'Add JIRA Server'.
3. A form will appear. Enter the information as follows:
   - 'Server Enabled' — Leave this checkbox ticked (default). If necessary, you can remove the tick to disable particular servers without deleting them. This is useful if your servers are behind a firewall and you don't have access to them.
   - 'Server Name' — A description of your JIRA server.
   - 'Server URL' — The address of your JIRA server.
   - 'Username' and 'Password' — The login name and password you use to access the JIRA server.
   - Remember Password — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.
   - If you choose to remember the password, it is stored in a Base64 encoding, so it is not really secure.
   - 'Use Default Credentials' — Put a tick in the checkbox if you want to use the single username and password that you have defined as your default credentials. You can set the default credentials on the 'Defaults' tab.
4. Click the 'Test Connection' button to check that the connection to the server works.
5. Click 'Apply' to save your changes and continue with server configuration, or 'OK' to save your changes and close the configuration tab.
6. Click the 'Defaults' tab to set your default server (if you have defined more than one JIRA server) and default project.
7. Now you can configure the JIRA options, as described below.

You can add more than one JIRA server,

Configuring your JIRA Options

1. Open the IDEA 'Settings' dialogue, then go to the 'IDE Settings' section and click the 'Atlassian Connector' icon.
2. Define the maximum number of issues that the connector will show on each screen. At display time, if there are more issues than specified here, the connector will display a 'Get More Issues...' link allowing you to retrieve the next batch of issues from the server.

RELATED TOPICS

Working with JIRA Issues in IDEA

_Configuring a JIRA Server in Visual Studio

Documentation under construction

Configuring your JIRA Server Connections

To configure your JIRA server connection(s):
1. Go to the 'Project Settings' for the Visual Studio Connector, by clicking the project configuration icon on your connector window.

To add a JIRA server:

1. Select the 'JIRA Servers' option in the left pane of the 'Project Configuration' window.
2. Click the 'Add New Server' button.
3. A form will display. Enter the information as follows:
   - 'Server Name' — A description of your JIRA server.
   - 'Server URL' — The address of your JIRA server.
   - 'Username' and 'Password' — The login name and password you use to access the JIRA server.
4. Click the 'Add Server' button to save your changes.
5. Click the 'Test Connection' button to check that the connection to the server works.
6. Click 'Close' to close the configuration tab.
7. Now you can configure the JIRA options, as described below.

You can add more than one JIRA server.

Configuring your JIRA Options

1. Go to the 'Global Settings' for the Visual Studio Connector, by clicking the configuration icon on your connector window.
2. Click the 'JIRA' tab. You will be able to configure the following options:
   - 'Issues Batch Size' — Define the maximum number of issues that the connector will show on each screen. At display time, if there are more issues than specified here, the connector will display a 'Get More Issues...' link allowing you to retrieve the next batch of issues from the server.

RELATED TOPICS

Working with JIRA Issues in IDEA

Configuring Task Repositories

1. In Eclipse Mylyn, open the 'Task Repositories' view:
   - Click 'Window', 'Show View', 'Other'.
   - Open the 'Tasks' category and select the 'Task Repositories' view.
2. Click the 'Add Task Repository' icon.
3. The 'Add Task Repository' screen appears, as shown below:

Now you can select and configure one or more of the task repositories and configure them to match your server(s), as described below.
Once you have set up a repository, it will appear in the Eclipse 'Task Repositories' view, as shown in this screenshot:

_Eclipse 3.3 Support_

End of support for Eclipse 3.3 (Europa)
Effective from version 1.3 of the Atlassian Connector for Eclipse, the connector no longer supports Eclipse 3.3 (Europa). We recommend that you upgrade your Eclipse platform to get the best benefit from Eclipse Mylyn and the Atlassian Connector for Eclipse.

_Flagship Image for Eclipse Bamboo_

Flagship image encapsulates functionality
The 'flagship' image should be the best representation of a particular set of functionality. This image will be used on an important page in the doc space which describes that functionality. The image will also be used in other spaces, such as the ATLAS space. This is a way to ensure that the images displayed in other spaces are kept up to date. The assumption is that the flagship image will be updated whenever necessary, as part of the normal documentation updates at each software release. (That's why the image should be used in a prominent place.

To use an image from this page:

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Creator</th>
<th>Creation Date</th>
<th>Comment</th>
</tr>
</thead>
</table>

_Flagship Image for IDEA Bamboo_

Flagship image encapsulates functionality
The 'flagship' image should be the best representation of a particular set of functionality. This image will be used on an important page in the doc space which describes that functionality. The image will also be used in other spaces, such as the ATLAS space. This is a way to ensure that the images displayed in other spaces are kept up to date. The assumption is that the flagship image will be updated whenever necessary, as part of the normal documentation updates at each software release. (That's why the image should be used in a prominent place.

To use an image from this page:

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Creator</th>
<th>Creation Date</th>
<th>Comment</th>
</tr>
</thead>
</table>

_Flagship Image for IDEA Crucible_
Flagship image encapsulates functionality
The 'flagship' image should be the best representation of a particular set of functionality. This image will be used on an important page in the doc space which describes that functionality. The image will also be used in other spaces, such as the ATLAS space. This is a way to ensure that the images displayed in other spaces are kept up to date. The assumption is that the flagship image will be updated whenever necessary, as part of the normal documentation updates at each software release. (That's why the image should be used in a prominent place.

To use an image from this page:

1. ![Flagship Image for IDEA Crucible](ImageName.png)

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Creator</th>
<th>Creation Date</th>
<th>Comment</th>
</tr>
</thead>
</table>

_Flagship Image for IDEA JIRA_

Flagship image encapsulates functionality
The 'flagship' image should be the best representation of a particular set of functionality. This image will be used on an important page in the doc space which describes that functionality. The image will also be used in other spaces, such as the ATLAS space. This is a way to ensure that the images displayed in other spaces are kept up to date. The assumption is that the flagship image will be updated whenever necessary, as part of the normal documentation updates at each software release. (That's why the image should be used in a prominent place.

To use an image from this page:

1. ![Flagship Image for IDEA JIRA](ImageName.png)

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Creator</th>
<th>Creation Date</th>
<th>Comment</th>
</tr>
</thead>
</table>

_Images_

Attached to this page are commonly-used images in the IntelliJ Connector documentation.

You can insert the images directly into a page — there's no need to attach them to the page itself. Here's an example of how to do that:

1. ![Images](MyImage.png)

Attachments

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Creator</th>
<th>Creation Date</th>
<th>Comment</th>
</tr>
</thead>
</table>
### Images for Release Notes

Attached to this page are commonly-used images in the IDE Plugin release notes.

You can insert the images directly into a page — there's no need to attach them to the page itself. Here's an example of how to do that:

```
![](Images for Release Notes.png)
```

#### Attachments

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Creator</th>
<th>Creation Date</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>7-sml.png</td>
<td>0.8 kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Aug 03, 2008 19:45</td>
<td></td>
</tr>
<tr>
<td>6-sml.png</td>
<td>0.8 kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Aug 03, 2008 19:45</td>
<td></td>
</tr>
<tr>
<td>3-sml.png</td>
<td>0.9 kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Aug 03, 2008 19:44</td>
<td></td>
</tr>
<tr>
<td>2-sml.png</td>
<td>0.9 kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Aug 03, 2008 19:44</td>
<td></td>
</tr>
<tr>
<td>1-sml.png</td>
<td>0.8 kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Aug 03, 2008 19:44</td>
<td></td>
</tr>
<tr>
<td>4-sml.png</td>
<td>0.8 kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Aug 03, 2008 19:44</td>
<td></td>
</tr>
<tr>
<td>5-sml.png</td>
<td>0.8 kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Aug 03, 2008 19:44</td>
<td></td>
</tr>
</tbody>
</table>
_Major Release Number Eclipse Connector

Version 2.0.x of the Atlassian Connector for Eclipse

_Major Release Number IntelliJ Connector

Version 2.2.x of the Atlassian Connector for IntelliJ IDEA

_Mylyn Introduction

The Atlassian Connector for Eclipse makes use of Mylyn's task-focused interface. Mylyn is shipped with the more recent versions of Eclipse. The Atlassian Connector for Eclipse requires Mylyn 3.2.x or Mylyn 3.3.x.

- If you do not already have Mylyn, it will be installed automatically when you install one of the Atlassian Connector's features.
- If you already have Mylyn installed, please check your version of Mylyn. If your version of Mylyn is earlier than 3.2, please upgrade to Mylyn 3.3.x.
- The Atlassian Connector for Eclipse now includes the JIRA Mylyn connector, which was previously available as a JIRA-only connector.
- The Mylyn documentation includes a quick start guide.

_Name of Eclipse Connector

New name: Atlassian Connector for Eclipse

We are changing the connector's name, from 'Atlassian Eclipse Connector' to 'Atlassian Connector for Eclipse'. The new name complies with the Eclipse Foundation's Guidelines for Eclipse Logos & Trademarks.

_Name of IDE Connector

_Name of IntelliJ Connector
The Atlassian Connector for IntelliJ IDEA stores configuration settings at two levels in IntelliJ IDEA:

- Server connections are stored as **project settings** in IntelliJ IDEA. Project settings allow you to share the same server connections with other members of your project team. Additionally, if you work on more than one project, this allows you to configure different servers for each project. Project-level settings can be stored in your source control repository, so that the connector will load the settings at the same time as loading the project into IDEA.
- Other options are stored as **IDE settings** in IntelliJ IDEA. IDE settings allow each developer to configure their own workspace-specific settings, such as polling intervals and the behaviour of notification popups.

**RELATED TOPICS**

**Configuring the IntelliJ Connector**

**Summary Bamboo in Eclipse**

- View a list of the builds you are monitoring, in the **Bamboo view in Eclipse**.
- Receive notification of failed builds and other build changes.
- Open the Bamboo build details in your Eclipse editor.
- Open the Bamboo build details in your web browser, displaying the Bamboo web interface.
- Run a build on the Bamboo server.
- View a Bamboo build log.
- View test results.
- View failed tests in the build.
- Comment on a Bamboo build.
- Label a Bamboo build.
- Add a new task based on a failed build.

**Summary Bamboo in IDEA**

- Receive notifications of failed builds.
- View the builds.
- Re-run a build.
- Open the Bamboo build details in an IDEA output tool window.
- View the build history for a selected plan.
- View a Bamboo build log.
- View failed tests and stack traces.
- Click a link in a stack trace to go directly to the code that failed.
- Re-run a failed test.
- View changed files.
- Compare the build version of a file with your local version.
- Compare the build version of a file with the previous repository version.
- Open the repository version of a file in your IDEA editor.
- Comment on a Bamboo build.
- Label a Bamboo build.

**Summary Bamboo in Visual Studio**

- View the builds.
- Re-run a build.
- Open the Bamboo build details in a browser.

**Summary Crucible in Eclipse**

- Add a query to retrieve and filter your reviews in Eclipse.
- View a list of your reviews in the Eclipse task list.
- Receive notification of new and updated reviews.
- View the review details in an Eclipse editor.
- Add, edit and reply to review comments in the review view or in the code editor.
- Open the file under review, at the commented source code line.
- Create a post-commit review, a patch review or a pre-commit review.
- Assign reviewers.
- Add changesets or patches to an existing review.
- Summarise and close the review and perform other workflow actions.
- Work on a review in the diff view.
- Work on a review in the review perspective.

_Summary Crucible in IDEA_

- Receive notifications of new and updated reviews.
- View your filtered reviews within IDEA.
- View the review details and comments in an IDEA output tool window.
- Create a post-commit review, a pre-commit review or a review during commit.
- View the source code that is under review.
- View the diff.
- Add a review comment from the comments tab or in the editor.
- View, edit and reply to comments in the source editor and in the diff view.
- Jump to the commented code in the source editor.
- Add a changelist to an existing review.
- Move a review through its workflow.
- Use the quick access options to open a review in IDEA.
- Open a review in the Crucible web interface from within IDEA.

_Summary JIRA in IDEA_

- View a filtered list of issues.
- Make a JIRA issue your active issue.
- Create a new JIRA issue.
- Comment on a JIRA issue and view existing comments.
- Create a changelist from a JIRA issue.
- Log work on a JIRA issue.
- View a JIRA issue in an IDEA output tool window.
- View stack traces from a JIRA issue and click through to the relevant source file.
- View and download attachments on an issue.
- Assign an issue to yourself or another user.
- Perform workflow actions on a selected issue.
- Use the issue quick access options to open an issue in IDEA.

_TOO MANY OPTIONS TO DOCUMENT_

Hint: There's more — too many options to document exhaustively here. Click around and try it out 😊

RELATED TOPICS

IDE Connector Documentation

_Version Compatibility for Eclipse Plugin_

The Atlassian Connector for Eclipse is compatible with the following software versions:

<table>
<thead>
<tr>
<th>Atlassian Connector for Eclipse</th>
<th>Bamboo</th>
<th>1.x</th>
<th>2.x recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crucible</td>
<td>1.6 or later</td>
<td>We recommend that you upgrade to the latest version of Crucible 1.6.x or 2.x for best results. However, please note that the new features of Crucible 2.0 are not reflected in the connector, such as iterative reviews, read/unread support, etc. Existing Crucible 1.6.x functionality will work when you connect to a Crucible 2.0 server.</td>
<td></td>
</tr>
<tr>
<td>Eclipse</td>
<td>3.4 (Ganymede)</td>
<td>3.5 (Galileo)</td>
<td></td>
</tr>
</tbody>
</table>
We recommend that you upgrade to the latest version of FishEye 1.6.x or 2.x for best results.

Ensure that JIRA SOAP/RPC services are running on the JIRA server (i.e. enable the RPC plugin).

The Atlassian Connector for Eclipse makes use of Mylyn’s task-focused interface. Mylyn is shipped with the more recent versions of Eclipse. The Atlassian Connector for Eclipse requires Mylyn 3.2.x or Mylyn 3.3.x.

- If you do not already have Mylyn, it will be installed automatically when you install one of the Atlassian Connector’s features.
- If you already have Mylyn installed, please check your version of Mylyn. If your version of Mylyn is earlier than 3.2, please upgrade to Mylyn 3.3.x.
- The Atlassian Connector for Eclipse now includes the JIRA Mylyn connector, which was previously available as a JIRA-only connector.
- The Mylyn documentation includes a quick start guide.

For Crucible reviews and FishEye source repository viewing, the connector integrates with Subversion using Subclipse or Subversive for source control and file retrieval.

Note: Install either Subclipse or Subversive integration option, not both
When selecting components of the Atlassian Connector for Eclipse, please select either Subclipse or Subversive, not both. (Installation details are below.) Installing one or the other of these two integration options will allow you to make full use of the Atlassian Connector’s integration with Crucible for code reviews and FishEye for source repository viewing, when versioning your project under SVN.

The connector provides limited support for CVS.

The connector does not currently support Perforce.
Although Crucible supports Perforce as a source repository, the Atlassian Connector for Eclipse does not currently support Perforce. This means that you will not be able to use the close integration between your Crucible reviews and your source directly in Eclipse. If you need it, please vote for this feature to help us prioritise the backlog.

Eclipse 3.3 (Europa) is no longer supported
Please note that, effective from version 1.3 of the Atlassian Connector for Eclipse, the connector will no longer support Eclipse 3.3 (Europa). We recommend that you upgrade your Eclipse platform to get the best benefit from Eclipse Mylyn and the Atlassian Connector for Eclipse.

Version Compatibility for IDEA Plugin
The Atlassian Connector for IntelliJ IDEA is compatible with the following software versions:

<table>
<thead>
<tr>
<th>Bamboo</th>
<th>1.x</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2.x recommended</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Crucible</th>
<th>1.6 or later</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Earlier versions of Crucible are not supported.</td>
</tr>
<tr>
<td></td>
<td>We recommend that you upgrade to the latest version of Crucible 1.6.x or 2.x for best results. However, please note that the new features of Crucible 2.0 are not reflected in the connector, such as iterative reviews, read/unread support, etc. Existing Crucible 1.6.x functionality will work when you connect to a Crucible 2.0 server.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FishEye</th>
<th>1.6 or later</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Earlier versions of FishEye are not supported.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>JIRA</th>
<th>3.7 or later</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3.12 or later recommended</td>
</tr>
</tbody>
</table>

| IntelliJ IDEA | 7.0.2 or later |

Version Compatibility for Visual Studio Connector
The Atlassian Connector for Visual Studio is compatible with the following software versions:

<table>
<thead>
<tr>
<th>Software</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>JIRA</td>
<td>3.7 or later&lt;br&gt;3.12 or later recommended</td>
</tr>
<tr>
<td>Microsoft Windows</td>
<td>XP or later – any version of Windows that is able to run Visual Studio 2008. &lt;br&gt;Note: You must have Microsoft .NET Framework 3.5 or later installed.</td>
</tr>
<tr>
<td>Microsoft Visual Studio</td>
<td>2008</td>
</tr>
</tbody>
</table>

___Watch this Space___

___What is the Plugin___

The Atlassian IDE Connectors are add-ons for your integrated development environment (IDE). They allow you to work with the Atlassian applications within your IDE. Now you don't have to switch between websites, email messages and news feeds to see what's happening to your project and your code. Instead, you can see the relevant JIRA issues, Crucible reviews and Bamboo build information right there in your development environment. You can also move quickly between your IDE and your FishEye view of your source repository.

___Icons___

This page is a library of icon images used in the plugin documentation.

**Note to documentation authors**

Use the icon images attached to this page, rather than attaching them to each individual page. That will make it easier to update the documentation if the plugin UI changes.

To use an image from this page:

```
!Icons^ImageName.png!
```

For example:

```
!Icons^BuildIcon.png!
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Creator</th>
<th>Creation Date</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eclipse-Crucible-AddRepositoryIcon.png</td>
<td>0.5  kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Mar 17, 2010 19:54</td>
<td></td>
</tr>
<tr>
<td>VSPollBuildsIcon.png</td>
<td>0.5  kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 25, 2010 22:55</td>
<td></td>
</tr>
<tr>
<td>VSViewBuildInBrowserIcon.png</td>
<td>0.6  kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 25, 2010 22:55</td>
<td></td>
</tr>
<tr>
<td>VSRunBuildIcon.png</td>
<td>0.3  kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 25, 2010 22:55</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-SelectRepositoryRe...</td>
<td>31   kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 18, 2010 23:08</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-ShowComment...</td>
<td>0.3  kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-PublishDraf...</td>
<td>0.5  kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Filename</td>
<td>Size</td>
<td>Author</td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>--------</td>
<td>----------------------</td>
<td>-----------</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-PrevCommentIcon.png</td>
<td>0.4 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-NextCommentIcon.png</td>
<td>0.4 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-LinkEditorIcon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-CollapseAllIcon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-ExpandAllIcon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-PublishCommentIcon.png</td>
<td>0.4 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-OpenActiveTaskIcon.png</td>
<td>0.5 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-MoreCommentActionsIcon.png</td>
<td>0.2 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-EditCommentIcon.png</td>
<td>0.4 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-DeleteCommentIcon.png</td>
<td>0.4 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-ReplyCommentIcon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-AddGeneralCommentIcon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-ShowUnreadIcon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-Review-AddFileCommentIcon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 17, 2010 19:01</td>
<td></td>
</tr>
<tr>
<td>Eclipse-Crucible-ActiveReviewIcon.png</td>
<td>0.2 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 16, 2010 18:41</td>
<td></td>
</tr>
<tr>
<td>VSEditFieldicon.png</td>
<td>0.4 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 07, 2010 18:38</td>
<td></td>
</tr>
<tr>
<td>VSEXPandAllicon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 05, 2010 00:04</td>
<td></td>
</tr>
<tr>
<td>VSCollapseAllicon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 05, 2010 00:04</td>
<td></td>
</tr>
<tr>
<td>VSAddCommentIcon.png</td>
<td>0.5 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 05, 2010 00:04</td>
<td></td>
</tr>
<tr>
<td>VSLogWorkicon.png</td>
<td>0.2 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 04, 2010 23:53</td>
<td></td>
</tr>
<tr>
<td>VSCloseIssueIcon.png</td>
<td>0.4 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 04, 2010 23:49</td>
<td></td>
</tr>
<tr>
<td>VSViewIssueActionsIcon.png</td>
<td>0.5 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 04, 2010 23:49</td>
<td></td>
</tr>
<tr>
<td>VSEditInBrowsericon.png</td>
<td>0.5 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 04, 2010 23:09</td>
<td></td>
</tr>
<tr>
<td>VSViewInBrowsericon.png</td>
<td>0.6 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 04, 2010 23:09</td>
<td></td>
</tr>
<tr>
<td>VSViewInIDEIcon.png</td>
<td>0.3 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 04, 2010 23:09</td>
<td></td>
</tr>
<tr>
<td>VSAddIssueIcon.png</td>
<td>0.6 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 04, 2010 23:09</td>
<td></td>
</tr>
<tr>
<td>VSFindIssueIcon.png</td>
<td>0.6 kB</td>
<td>Andrew Lui [Atlassian Technical Writer]</td>
<td>Feb 04, 2010 23:09</td>
<td></td>
</tr>
<tr>
<td>Icon Name</td>
<td>Size</td>
<td>Author</td>
<td>Date</td>
<td>Time</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-------</td>
<td>-----------------------------------------</td>
<td>------------</td>
<td>----------</td>
</tr>
<tr>
<td>CommentIcon.png</td>
<td>0.3kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Feb 04, 2009</td>
<td>21:31</td>
</tr>
<tr>
<td>IDEAAddIssueIcon.png</td>
<td>0.6kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Feb 01, 2009</td>
<td>21:27</td>
</tr>
<tr>
<td>IDEAAddIssueIcon.png</td>
<td>0.6kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Feb 01, 2009</td>
<td>21:27</td>
</tr>
<tr>
<td>IDEAAddIssueIcon.png</td>
<td>0.6kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Feb 01, 2009</td>
<td>21:27</td>
</tr>
<tr>
<td>IDEAAddIssueIcon.png</td>
<td>0.6kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Feb 01, 2009</td>
<td>21:27</td>
</tr>
<tr>
<td>IDEAAddIssueIcon.png</td>
<td>0.6kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Feb 01, 2009</td>
<td>21:27</td>
</tr>
<tr>
<td>IDEAAddIssueIcon.png</td>
<td>0.6kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Feb 01, 2009</td>
<td>21:27</td>
</tr>
<tr>
<td>IDEAAddIssueIcon.png</td>
<td>0.6kB</td>
<td>Sarah Maddox [Atlassian Technical Writer]</td>
<td>Feb 01, 2009</td>
<td>21:27</td>
</tr>
</tbody>
</table>
This page is a container for all the bookmarks in this space. Do not delete or move it or you will lose all your bookmarks.

The 15 most recent bookmarks in Atlassian IDE Connectors

There are no bookmarks to display.
Introduction to the Atlassian IDE Connectors

The Atlassian IDE Connectors are add-ons for your integrated development environment (IDE). They allow you to work with the Atlassian applications within your IDE. Now you don't have to switch between websites, email messages and news feeds to see what's happening to your project and your code. Instead, you can see the relevant JIRA issues, Crucible reviews and Bamboo build information right there in your development environment. You can also move quickly between your IDE and your FishEye view of your source repository.

The Atlassian IDE Connectors are available for IntelliJ IDEA, Eclipse and Visual Studio (beta). Take a look at the documentation for each IDE:

- Atlassian Connector for Eclipse
- Atlassian Connector for IntelliJ IDEA
- Atlassian Connector for Visual Studio (beta)

Visual Studio Support
We are currently developing an Atlassian IDE Connector for Microsoft Visual Studio. If you are interested in Visual Studio support, please follow the project's progress on our issue tracker. You can create an account and add feature requests, comments and bug reports to help determine our roadmap.

Overview of the Atlassian IDE Connectors

The Atlassian IDE Connectors are add-ons for your integrated development environment (IDE). They allow you to work with the Atlassian applications within your IDE. Now you don't have to switch between websites, email messages and news feeds to see what's happening to your project and your code. Instead, you can see the relevant JIRA issues, Crucible reviews and Bamboo build information right there in your development environment. You can also move quickly between your IDE and your FishEye view of your source repository.

The Atlassian IDE Connectors are available for IntelliJ IDEA, Eclipse and Visual Studio (beta). Take a look at the documentation for each IDE:

- Atlassian Connector for Eclipse
- Atlassian Connector for IntelliJ IDEA
- Atlassian Connector for Visual Studio (beta)

Visual Studio Support
We are currently developing an Atlassian IDE Connector for Microsoft Visual Studio. If you are interested in Visual Studio support, please follow the project's progress on our issue tracker. You can create an account and add feature requests, comments and bug reports to help determine our roadmap.

Atlassian Connector for Eclipse

The connector does not currently support Perforce. Although Crucible supports Perforce as a source repository, the Atlassian Connector for Eclipse does not currently support Perforce. This means that you will not be able to use the close integration between your Crucible reviews and your source directly in Eclipse. If you need it, please vote for this feature to help us prioritise the backlog.

Version 2.0.x of the Atlassian Connector for Eclipse

Installation Guide
The Installation Guide is for people who are installing the Atlassian Connector for Eclipse for the first time. Check the system requirements and install the connector. You may also find the Configuration Guide and FAQ useful.

Upgrade Guide
The Upgrade Guide is for people who are upgrading their connector to the latest version.

JIRA User’s Guide
The connector's JIRA User Guide is for developers who want to monitor, add and update JIRA issues from within Eclipse. Learn how to configure the connector and work with JIRA issues in Eclipse.

Bamboo User’s Guide
The connector's Bamboo User Guide is for developers who want to use Bamboo build information from within Eclipse. Learn how to
configure the connector and work with Bamboo builds in Eclipse.

**FishEye User's Guide**

The connector's FishEye User Guide is for developers who want to move quickly between Eclipse and the FishEye view of their source repository. Learn how to configure the connector and work with your FishEye repository view in Eclipse.

**Crucible User's Guide**

The connector's Crucible User Guide is for developers who want to conduct Crucible code reviews from within Eclipse. Learn how to configure the connector and work with Crucible reviews in Eclipse.

**Developer Resources**

This is an open source project. To access the connector's source code, point your SVN at: https://studio.atlassian.com/svn/PLE/trunk. If you do not already have an account, please go to http://studio.atlassian.com and sign up for an account. You should then be able to access the svn repository with your new account. You may also find the Atlassian IDE Connector Forum useful.

**Installation and Upgrade Guide for the Eclipse Connector**

- License and Copyright for the Eclipse Connector
- Installing the Eclipse Connector
- Upgrading the Eclipse Connector
- Upgrading from the JIRA Connector to the Atlassian Connector for Eclipse
- Atlassian Connector for Eclipse - Release Notes

**License and Copyright for the Eclipse Connector**

**Open Source**

This is an open source project. You can get the source code from our SVN repository.

**Copyright Statement**

© Atlassian 2008

**License**

Licensed under the Eclipse Public License, Version 1.0 (the "License").

You may not use this file except in compliance with the License.

You may obtain a copy of the License at


Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

**Installing the Eclipse Connector**

Below are the system requirements and installation instructions for the Atlassian Connector for Eclipse.

**On this page:**

- System Requirements
- Eclipse Update Sites
- Offline Installation
- Installation for Eclipse 3.5 (Galileo)
- Installation for Eclipse 3.4 (Ganymede)
- Troubleshooting the Installation

**System Requirements**

The Atlassian Connector for Eclipse is compatible with the following software versions:
### Atlassian Connector for Eclipse

<table>
<thead>
<tr>
<th>Application</th>
<th>Version</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bamboo</td>
<td>1.x</td>
<td>Bamboo 1.x recommended.</td>
</tr>
<tr>
<td></td>
<td>2.x recommended</td>
<td></td>
</tr>
<tr>
<td>Crucible</td>
<td>1.6 or later</td>
<td>We recommend that you upgrade to the latest version of Crucible 1.6.x or 2.x for best results. However, please note that the new features of Crucible 2.0 are not reflected in the connector, such as iterative reviews, read/unread support, etc. Existing Crucible 1.6.x functionality will work when you connect to a Crucible 2.0 server.</td>
</tr>
<tr>
<td>Eclipse</td>
<td>3.4 (Ganymede) 3.5 (Galileo)</td>
<td></td>
</tr>
<tr>
<td>FishEye</td>
<td>1.6 or later</td>
<td>We recommend that you upgrade to the latest version of FishEye 1.6.x or 2.x for best results.</td>
</tr>
<tr>
<td>JIRA</td>
<td>3.4 or later 3.7 or later recommended</td>
<td>Ensure that JIRA SOAP/RPC services are running on the JIRA server (i.e. enable the RPC plugin)</td>
</tr>
<tr>
<td>Mylyn</td>
<td>The Atlassian Connector for Eclipse makes use of Mylyn's task-focused interface. Mylyn is shipped with the more recent versions of Eclipse. The Atlassian Connector for Eclipse requires Mylyn 3.2.x or Mylyn 3.3.x.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• If you do not already have Mylyn, it will be installed automatically when you install one of the Atlassian Connector's features.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• If you already have Mylyn installed, please check your version of Mylyn. If your version of Mylyn is earlier than 3.2, please upgrade to Mylyn 3.3.x.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The Atlassian Connector for Eclipse now includes the JIRA Mylyn connector, which was previously available as a JIRA-only connector.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The Mylyn documentation includes a quick start guide.</td>
<td></td>
</tr>
<tr>
<td>Subversion</td>
<td>For Crucible reviews and FishEye source repository viewing, the connector integrates with Subversion using Subclipse or Subversive for source control and file retrieval.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note: Install either Subclipse or Subversive integration option, not both</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>When selecting components of the Atlassian Connector for Eclipse, please select either Subclipse or Subversive, not both. (Installation details are below.) Installing one or the other of these two integration options will allow you to make full use of the Atlassian Connector’s integration with Crucible for code reviews and FishEye for source repository viewing, when versioning your project under SVN.</td>
<td></td>
</tr>
<tr>
<td>CVS</td>
<td>The connector provides limited support for CVS.</td>
<td></td>
</tr>
<tr>
<td>Perforce</td>
<td><strong>The connector does not currently support Perforce.</strong> Although Crucible supports Perforce as a source repository, the Atlassian Connector for Eclipse does not currently support Perforce. This means that you will not be able to use the close integration between your Crucible reviews and your source directly in Eclipse. If you need it, please vote for this feature to help us prioritise the backlog.</td>
<td></td>
</tr>
<tr>
<td>Eclipse 3.3 (Europa)</td>
<td></td>
<td>Eclipse 3.3 (Europa) is no longer supported</td>
</tr>
</tbody>
</table>

Please note that, effective from version 1.3 of the Atlassian Connector for Eclipse, the connector will no longer support Eclipse 3.3 (Europa). We recommend that you upgrade your Eclipse platform to get the best benefit from Eclipse Mylyn and the Atlassian Connector for Eclipse.

### Eclipse Update Sites

Add the following URL to your Eclipse update sites:

- For Eclipse 3.4 and 3.5 — [http://update.atlassian.com/atlassian-eclipse-plugin/e3.4](http://update.atlassian.com/atlassian-eclipse-plugin/e3.4)
- For Eclipse 3.3 (Not supported. The last compatible version of the connector is 1.2.0) — [http://update.atlassian.com/atlassian-eclipse-plugin/1.2.0/e3.3](http://update.atlassian.com/atlassian-eclipse-plugin/1.2.0/e3.3)

There is also a weekly update site available, for those who are keen to get the latest version of the connector. Builds are executed every Sunday at approximately 2am:

- For Eclipse 3.4 and 3.5 — [http://update.atlassian.com/atlassian-eclipse-plugin/weekly/e3.4/site.xml](http://update.atlassian.com/atlassian-eclipse-plugin/weekly/e3.4/site.xml)
Offline Installation

If you cannot use our update site (e.g., due to firewall issues) or just prefer local installation, we offer a zipped version of the entire update site, including the Mylyn dependencies.

1. First install the Subclipse or Subversive plugin for Eclipse, if you version your project under SVN and if you do not already have one of these plugins installed in your IDE. This will allow you to take advantage of the connector’s Crucible and FishEye integration.
2. Download the Atlassian connector’s archived update site.
4. Continue with the installation steps as described for the regular update site — see instructions below.

⚠️ Please note: In the zipped update site we do not include Subclipse or Subversive. You will need to install them separately before installing Atlassian Connector, if you want to take advantage of the connector’s Crucible and FishEye integration.

Installation for Eclipse 3.5 (Galileo)

If you are using Eclipse 3.5 (Galileo), you can install the connector via the Eclipse software updates manager, as described below.

To install the connector on a clean Eclipse 3.5 (Galileo) IDE for Java Developers:

1. Open the Eclipse software updates manager via ‘Help,’ ‘Install New Software’.
2. Click ‘Add,’ enter the details of the update site, as shown in the screenshot below:

   ![Add Site](image)

3. Click ‘OK.’
5. Select both ‘Atlassian Connector’ categories. Also select either ‘Subclipse’ or ‘Subversive’, depending on which of the two you are using, as shown in the screenshot below:
6. Click 'Next' and follow the installation prompts.
7. Restart Eclipse when prompted.

**Installation for Eclipse 3.4 (Ganymede)**

The steps below show the full installation procedure in Eclipse 3.4 (Ganymede):

1. If you already have Mylyn installed:
   - Check your version of Mylyn as described under dependencies above.
   - Open the Eclipse software updates manager via 'Help', 'Software Updates', 'Installed Software' and click 'Update' to get the latest Mylyn updates.
   - Restart Eclipse.

2. Open the Eclipse software updates manager via 'Help', 'Software Updates', 'Available Software'.

3. Click the 'Manage Sites' button.

4. Scroll down to find the 'Mylyn Extras' update site somewhere near the bottom, enable it and click 'OK'.

5. Back on the 'Available Software' tab, click the 'Add Site' button.

6. The 'Add Site' dialogue box will appear, as shown in the screenshot below:

   ![Add Site](image)

   Enter the URL of the Atlassian Connector update site:
   http://update.atlassian.com/atlassian-eclipse-plugin/e3.4

7. Click 'OK'.

8. The 'Atlassian Connector for Eclipse' will appear in the list of available software, as shown in the screenshot below:
9. Select the required features for the connector:
   - Atlassian Connector for Eclipse — Required. This is the core of the Atlassian Connector.
   - Bamboo Integration — Allows you to view and manage your Bamboo builds within your Eclipse IDE.
   - Crucible and FishEye Integration — Allows you to view and manage your Crucible code reviews within your Eclipse IDE, and to move easily to a FishEye view of your repository.
   - CVS Integration — Select this integration option if you version your project under CVS.
   - Subclipse Integration — Select this integration option if you version your project under SVN with Subclipse.
   - Subversive Integration — Select this integration option if you version your project under SVN with Subversive.
   - The installation process will automatically provide the connector's JIRA integration, allowing you to view and manage your JIRA issues within your Eclipse IDE.

10. Click the 'Install' button, then click 'Finish' when asked to confirm the installation.

11. If prompted, accept the license agreement and click the 'Next' button.

12. Restart Eclipse when prompted.

13. The Atlassian Connector for Eclipse has now been installed. Your next step is to add your Bamboo, Crucible and JIRA task repositories in Eclipse. See Configuring the Eclipse Connector.

Troubleshooting the Installation

Please refer to the list of installation FAQ:

- Error while Collecting Items to be Installed
- Installation Error - Cannot Find a Solution Satisfying the Following Requirements

RELATED TOPICS

- Configuring the Eclipse Connector
- License and Copyright for the Eclipse Connector
- Installing the Eclipse Connector
- Upgrading the Eclipse Connector
- Upgrading from the JIRA Connector to the Atlassian Connector for Eclipse
- Atlassian Connector for Eclipse - Release Notes

IDE Connector Documentation

Limited Support for Specific SCMs

Currently the Atlassian Connector for Eclipse provides full support for code reviews where you version your project under SVN using Subclipse or Subversive. We provide limited support for CVS, as described below.
We appreciate your feedback
If you are using CVS, please let us know of any problems or give us other feedback. You can raise issues and requests on our issue tracker.

On this page:
- Full Support for SVN with Subclipse or Subversive
- Limited Support for CVS
- Other SCMs not yet Supported

**Full Support for SVN with Subclipse or Subversive**

The Atlassian Connector for Eclipse provides full support for Subclipse or Subversive where you version your project under SVN. For a list of supported features, please refer to the version 1.0 and version 1.1 of the Atlassian Connector for Eclipse.

- Only Subversive 0.7.8 or later is supported.

**Limited Support for CVS**

The Atlassian Connector for Eclipse provides limited support for Crucible code reviews where you version your project under CVS:

- You can open review files and add comments.
- You cannot add new reviews or changsets.
- You cannot perform pre-commit reviews.

The connector's FishEye functionality provides full support for CVS.

**Other SCMs not yet Supported**

We intend to add support for other SCMs via the connector's integration with Crucible, so that the connector will support the SCMs which Crucible supports.

- The connector does not currently support Perforce. Although Crucible supports Perforce as a source repository, the Atlassian Connector for Eclipse does not currently support Perforce. This means that you will not be able to use the close integration between your Crucible reviews and your source directly in Eclipse. If you need it, please vote for this feature to help us prioritise the backlog.

**RELATED TOPICS**

- Installing the Eclipse Connector
- IDE Connector Documentation

**Upgrading the Eclipse Connector**

This page tells you how to upgrade the Atlassian Connector for Eclipse. If you have never installed the connector, please refer to the installation guide instead.

**Replacing Early Versions of the ‘Atlassian IDE Plugin’**

If you have previously installed version 0.4 or earlier of the Atlassian IDE Plugin:

1. Uninstall the earlier version.
2. Then follow the installation guide to install the new Atlassian Connector for Eclipse.

**Upgrading from Version 1.0 or Later**

If you are upgrading from version 1.0 Beta, version 1.0 or later versions of the Atlassian Connector for Eclipse, you can follow the standard Eclipse software upgrade procedure.

These are the steps in the standard upgrade procedure for Eclipse 3.4 (Ganymede):

2. The ‘Installed Software’ screen will appear, as shown in the screenshot below:
Select the required components of the 'Atlassian Connector for Eclipse' and click the 'Update' button.

3. When prompted, verify the installation details and click the 'Next' button.

4. If prompted, accept the license agreement and click the 'Finish' button.

5. When prompted, restart Eclipse.

RELATED TOPICS

- Configuring the Eclipse Connector
- License and Copyright for the Eclipse Connector
- Installing the Eclipse Connector
- Upgrading the Eclipse Connector
- Upgrading from the JIRA Connector to the Atlassian Connector for Eclipse
- Atlassian Connector for Eclipse - Release Notes

IDE Connector Documentation

Upgrading from the JIRA Connector to the Atlassian Connector for Eclipse

If you have already installed the JIRA Connector via the Mylyn Connector Discovery in Eclipse, only the JIRA component of the Atlassian Connector for Eclipse will be installed. Follow the instructions below to upgrade to the full Atlassian Connector for Eclipse, which includes FishEye, Crucible and Bamboo integration.

If you have never installed the connector, please refer to the installation guide instead. If you want to upgrade an existing installation of the Atlassian Connector for Eclipse, please refer to this upgrade guide instead.

On this page:

- Offline Upgrade
- Upgrade for Eclipse 3.5 (Galileo)
- Upgrade for Eclipse 3.4 (Ganymede)

Offline Upgrade

If you are upgrading offline, follow the instructions for installing the connector offline.

Upgrade for Eclipse 3.5 (Galileo)

To upgrade the JIRA connector to the Atlassian Connector for Eclipse on Eclipse 3.5 (Galileo) IDE:

1. Open the Eclipse software updates manager via 'Help', 'Install New Software'.
   - The Atlassian Connector update site will have been added automatically when you installed the JIRA connector. If it hasn't been added, you can add it manually yourself (see Installing the Eclipse Connector for the appropriate URL).

2. Select the Atlassian Connector update site from the 'Work with' dropdown. The 'Atlassian Connector', 'Atlassian Integrations' and 'Dependencies' items will display as shown in the screenshot below.
3. Select the required features for the connector:
   - **Atlassian Connector for Eclipse** — Required. This is the core of the Atlassian Connector.
   - **Atlassian Bamboo Integration** — Allows you to view and manage your Bamboo builds within your Eclipse IDE.
   - **Atlassian Crucible & FishEye Integration** — Allows you to view and manage your Crucible code reviews within your Eclipse IDE, and to move easily to a FishEye view of your repository.
   - **Atlassian CVS Integration** — Select this integration option if you version your project under CVS. If you select this, do not select 'Atlassian Subversive Integration'.
   - **Atlassian Subclipse Integration** — Select this integration option if you version your project under SVN with Subclipse. If you select this, do not select 'Atlassian Subversive Integration'.
   
4. Click the 'Next' button and follow the installation prompts.
5. Restart Eclipse when prompted.
6. The Atlassian Connector for Eclipse has now been installed. Your next step is to add your Bamboo, Crucible and JIRA task repositories in Eclipse. See **Configuring the Eclipse Connector**.

**Upgrade for Eclipse 3.4 (Ganymede)**

To upgrade the JIRA connector to the Atlassian Connector for Eclipse on Eclipse 3.4 (Ganymede) IDE:

1. Open the Eclipse software updates manager via 'Help', 'Software Updates', 'Available Software'.
   
   The Atlassian Connector update site will have been added automatically when you installed the JIRA connector. If it hasn't been added, you can add it manually yourself (see **Installing the Eclipse Connector** for the appropriate URL).
2. The 'Atlassian Connector for Eclipse' will appear in the list of available software, as shown in the screenshot below:
Select the required features for the connector:

- **Atlassian Connector for Eclipse** — Required. This is the core of the Atlassian Connector.
- **Atlassian Bamboo Integration** — Allows you to view and manage your Bamboo builds within your Eclipse IDE.
- **Atlassian Crucible and FishEye Integration** — Allows you to view and manage your Crucible code reviews within your Eclipse IDE, and to move easily to a FishEye view of your repository.
- **Atlassian CVS Integration** — Select this integration option if you version your project under CVS.
- **Atlassian Subclipse Integration** — Select this integration option if you version your project under SVN with Subclipse. If you select this, do not select 'Atlassian Subversive Integration'.
- **Atlassian Subversive Integration** — Select this integration option if you version your project under SVN with Subversive. If you select this, do not select 'Atlassian Subclipse Integration'.

Click the 'Install' button, then click 'Finish' when asked to confirm the installation.

Click the 'Next' button and follow the installation prompts.

Restart Eclipse when prompted.

The Atlassian Connector for Eclipse has now been installed. Your next step is to add your Bamboo, Crucible and JIRA task repositories in Eclipse. See Configuring the Eclipse Connector.

**Atlassian Connector for Eclipse - Release Notes**

- **New name: Atlassian Connector for Eclipse**
  We are changing the connector's name, from 'Atlassian Eclipse Connector' to 'Atlassian Connector for Eclipse'. The new name complies with the Eclipse Foundation's Guidelines for Eclipse Logos & Trademarks.

- **Current released version:**
  Version 2.0 Beta of the Atlassian Connector for Eclipse has now been released — see the release notes.

**Atlassian Connector for Eclipse - v2.0 Release Notes**

17 March 2010
With pleasure, Atlassian presents version 2.0 of the Atlassian Connector for Eclipse. This release contains the same features that we previewed in the v2.0 Beta release plus more.

We've overhauled Crucible reviews in your Eclipse IDE this release. The new Crucible Review perspective allows you to work on the review as well as the file content at the same time, without having to switch back and forth between the two. The Crucible Review perspective also organises your workspace into a logical layout that makes working with reviews even easier.

You can now create post-commit Crucible reviews from multiple files within your Eclipse IDE, via the Package Explorer, or from a Java file viewed in the editor. Files selected in the Package Explorer can also be added to active reviews. The connector also supports additional SCM systems for post-commit reviews. Post-commit reviews created from changesets now support all SCMs currently supported by Crucible.

This version also includes annotations support for non-Java files (e.g. C++, PHP, Scala, Rails, Flex and more) and wiki markup support for review comments.

There are a number of smaller improvements and bug fixes, listed below.

---

**Important Upgrade Notes**

The Atlassian Connector for Eclipse - v2.0 Upgrade Notes contains important information that you must read before upgrading.

---

**Don't have the Atlassian Connector for Eclipse yet?**

You can install the connector directly from the Eclipse software updates manager or from a zipped archive. Follow our installation guide.

---

**Highlights of this Release:**

- Crucible Review Perspective
- Improvements to Crucible Review Creation
- Addition of Files to Reviews
- Extended Annotations Support
- Wiki Markup Support for Comments
- Support for Additional SCM Systems for Post-Commit Reviews
- Other Things Worth Mentioning

🌟 We love your feedback! 🌟

Please log your issues, requests and votes. They help us decide what needs doing.

This is an open source project. You can get the source code from our SVN repository.

---

**Highlights of this Release**

1. **Crucible Review Perspective**

   The new Crucible Review perspective allows you to work on the review as well as the file content at the same time. You can navigate, view/add/remove comments and more; all without having to switch back and forth between the two. The Crucible Review perspective also organises your workspace into a logical layout that makes working with reviews even easier.

   The new Crucible Review perspective also presents you with a workspace layout that makes working with reviews even easier. Key information about the review is displayed in a new Review Explorer in this perspective. The Review Explorer also keeps all of the commonly used review functions at your fingertips. Finally, the new perspective also includes a new Review Comment view. This view provides you with the information and functions you need to deal with individual review comments, all in one place.
Improvements to Crucible Review Creation

We’ve added a number of new ways for you to create reviews in your Eclipse IDE. Post-commit reviews can now be created from files selected in the Package Explorer or from the Java file you are currently viewing in the Active Task Editor. You can also add a file to an active review via the Package Explorer.
Addition of Files to Reviews

In addition to new ways to create reviews in your Eclipse IDE, you can now also add files to reviews.

Extended Annotations Support

The Atlassian Connector for Eclipse now supports Crucible annotations for any non-Java file. This includes files from a whole range of technologies, including C++, PHP, Scala, Rails, Flex and more.
Wiki Markup Support for Comments

We’ve added wiki markup support for comments in this release. Comments that use wiki markup will be rendered appropriately in your IDE. The connector also includes a robust wiki markup editor for comments that provides auto-completion for wiki markup notation, issue keys and words; suggests spelling corrections; and offers two preview modes (simple and browser).
Support for Additional SCM Systems for Post-Commit Reviews

The connector now supports all SCM systems supported by Crucible for post-commit reviews created from changesets. For information on the SCM systems supported by Crucible, i.e. Subversion, CVS, Perforce, Git and IBM Clearcase. Please see Crucible Supported Platforms for more information.

Other Things Worth Mentioning

This release brings a number of bug fixes and improvements. The following items are worth a special mention:

- Publish all draft comments with a single click.
- Create ad-hoc reviews directly from your code.
- New functions for working with comments — next/previous comment navigation, filter out read comments, mark comments as read/unread, reply to replies.
- Support for working on reviews without moderators (note, creation of reviews without moderators is not supported yet).
- Pre-defined filters in the New Query wizard for JIRA.
- Restrict JIRA comment visibility to specific roles.

Complete List of Fixes in this Release
<table>
<thead>
<tr>
<th>Key</th>
<th>Summary</th>
<th>Priority</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLE-1041</td>
<td>Can't cancel mark as read job in Comment View after refresh</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1039</td>
<td>Annotations not visible for local file opened from Review View if it was already open in the editor</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1038</td>
<td>[regression] problems in Review Perspective</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1037</td>
<td>Ability to refresh review from Review View</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1028</td>
<td>Creating review with comment from changed file (editor) does not add a comment</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-1025</td>
<td>Comments added to the compare editor are not visible</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-1024</td>
<td>Problem when opening file from Review Explorer.</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-1022</td>
<td>Review Explorer synchronize with editor problem</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-1019</td>
<td>Delay marking the comment as read in Review Explorer / Comment View</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1018</td>
<td>Support Wiki markup when adding crucible comment</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1017</td>
<td>Unread comments filter should not hide comments</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1011</td>
<td>Review Explorer focus problem</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-1008</td>
<td>Eclipse connector is unable to log in to latest Studio</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1003</td>
<td>Implement new usage reports on our wiki</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-1001</td>
<td>jira query wizard does not allow to combine &quot;no component&quot; with other concrete components for 'in components' filter</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-1000</td>
<td>jira query wizard does not allow to combine &quot;no version&quot; with other concrete versions for 'reported in' filter</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-999</td>
<td>NPE in Create Review wizard (Back button does not work)</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-997</td>
<td>[regression] Add to Active review does not work from the editor - it wants to create a new review instead</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-993</td>
<td>Reviewers list broken in review details window (no display name)</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-990</td>
<td>People installing just JIRA Connector should get our Support (StAC) JIRA server added automatically as a Task Repository</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-988</td>
<td>[installation] Bamboo integration is installed when the user just selects Atlassian Connector for Eclipse (Required) feature</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-987</td>
<td>Show referenced file revisions for a single comment in Comment View</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-986</td>
<td>Support wiki markup in Crucible review comments and statement of objectives</td>
<td>✅</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-984</td>
<td>Mylyn JIRA connector hyperlink detector doesn't always correctly detect JIRA issue hyperlinks</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-983</td>
<td>New comment part error when creating new task</td>
<td>🚧</td>
<td>Closed</td>
</tr>
<tr>
<td>JIRA Key</td>
<td>Description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-982</td>
<td>Synchronize problems in review perspective</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-979</td>
<td>Double clicking in Review Explorer on a comment to changed XML file opens this file, but moves the selection from clicked node to the file node</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-977</td>
<td>Sort line comments in Review Explorer tree by lines first, not just by dates</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-976</td>
<td>Tooltip for file nodes in Review Explorer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-975</td>
<td>Link with editor setting in Review Explorer is not saved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-974</td>
<td>SWTError in Review Perspective when linking is on</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-973</td>
<td>[regression] invalid compacting of review tree in cases when directories themselves are review items</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-972</td>
<td>Show revision string in file name tab (editor) for resources fetched externally while doing review</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-970</td>
<td>Atlassian Connector features (and logo) are not displayed after upgrading just Mylyn 3.3.x to Mylyn 3.3.99</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-968</td>
<td>Widget is disposed exception</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-966</td>
<td>[usage reporting] Add info about which Atlassian features are installed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-952</td>
<td>Selection (and Comment View) is not correctly refreshed when a comment is changed (e.g. due to update or publishing)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-946</td>
<td>Show reply in review explorer tree after it's has just been added by the user</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-944</td>
<td>Align reviewers list widget with objectives/comment text box in the last page of create review wizard.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-943</td>
<td>Next/Prev comment action does not sync with Comment view</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-940</td>
<td>Tree view in Review Wizard shows multiple entries for the same path</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-939</td>
<td>Get repository changes directly from Crucible using new REST API</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-938</td>
<td>Separate 'objectives' and 'comment' text controls when creating review from selection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-937</td>
<td>Select/unselect with spacebar does not work as expected.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-936</td>
<td>Allow to create pre-commit review if there is no connector for selected files (for create review action)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-935</td>
<td>Ability to add general comment from Review Explorer View</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-932</td>
<td>fisheye integration does network I/O in UI thread when right clicking resource</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-930</td>
<td>[regression] 3.3.0 breaks Update Attributes From Repository on JIRA 3.6 and earlier</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-929</td>
<td>investigate possible race condition in JiraRepositoryConnectorTest.testMarkStaleClosedTask</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-925</td>
<td>provide UI for setting comment visibility</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-913</td>
<td>Green Hopper ranking custom field is erased on submit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Description</td>
<td>Status</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>PLE-908</td>
<td>Query with &quot;assigned to&quot; field set to unassigned is ignored</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-907</td>
<td>when editing jira task, it removes attributes from the issue, specifically the &quot;Fix Version/s&quot; field</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-905</td>
<td>jira query wizard don't allow to combine &quot;fix for&quot; &quot;no fix version&quot; and other concrete versions</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-891</td>
<td>reassigning an issue and stopping progress in the same 'submit' causes undesirable results</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-874</td>
<td>Do not allow posting non-draft replies to drafts</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-873</td>
<td>Move general comments to Review Explorer view</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-872</td>
<td>Opening a file or a comment from Review Explorer should automatically show Review Comments view</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-871</td>
<td>Merge Create Review from selected resource(s) and Discuss code actions into one action</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-870</td>
<td>Display all Crucible servers in Task Repository selection wizard page and disable those which are not capable of handling given request</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-869</td>
<td>Add a tool-tip for pre-commit entries in Create Review wizard explaining what pre-commit means here</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-868</td>
<td>Remove &quot;post-commit&quot; decoration from Create Review wizard</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-867</td>
<td>Enhance error handling</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-864</td>
<td>Widget is disposed bug</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-862</td>
<td>NPE when playing with Crucible perspective</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-860</td>
<td>Open remote files using Crucible 2.x API</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-859</td>
<td>Wrong url used to validate mappings for Subversion</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-856</td>
<td>Display general comments in Review Explorer</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-855</td>
<td>Use icon decorations for modified/added/removed state in Review Explorer instead of decoration</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-854</td>
<td>Navigate next/prev comment in Review Explorer</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-851</td>
<td>use colours in Review Explorer</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-850</td>
<td>Comments marked as defects are not marked in Review Explorer</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-849</td>
<td>Synchronize Crucible Explorer View and Crucible Comments View with Editor</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-848</td>
<td>Add show only unread comments filter to Review Explorer View</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-847</td>
<td>Use images in Crucible Explorer View</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-846</td>
<td>Compact entries in Crucible Explorer View</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-845</td>
<td>Handle double-click in Crucible Explorer View</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-844</td>
<td>Add Expand All / Collapse All toolbar actions in Crucible Explorer View</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-842</td>
<td>Add shortcut icon for review editor in Explorer and Comments views</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-841</td>
<td>Having wrong mapping gives me HTTP400 from Crucible</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-840</td>
<td>Facilitating installation of Connector when only JIRA Connector (listed on Mylyn Discovery) is installed</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-839</td>
<td>Empty error message in Error Log view</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-837</td>
<td>Show test results action doesn't compile on e3.6</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-834</td>
<td>Analyse inconsistency in recent usage reports and fix them</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-807</td>
<td>Improve explanation why Repository mappings are needed and how to set them up</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-806</td>
<td>Do not automatically propose a mapping (add mapping dialog) when accessing the mapping dialog via Define Repository Mappings from Select Changeset wizard page when all required mappings are present</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-804</td>
<td>Merge Create Pre-commit and Post-commit Review actions into one action which is capable of handling mixed reviews (with post-commit items and pre-commit items together)</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-803</td>
<td>Do not ask user whether to open Bamboo view (after Bamboo task repo is added) and just focus on this view.</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-802</td>
<td>Support dirty files in &quot;Add files to active review&quot; - add them in pre-commit mode</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-801</td>
<td>In create review from changeset page: display which changeset (from which SCM) does not have correct mapping configured</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-799</td>
<td>Add separator in Team menu between FishEye and Crucible related actions</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-797</td>
<td>Support reviews without the moderator (introduced with Crucible 2.2M1)</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-796</td>
<td>Use Crucible/FishEye mappings when opening files from Crucible Review</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-793</td>
<td>Cannot create post-commit review if selection contains svn-ignored file</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-791</td>
<td>Duplicated entries in pre-commit review</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-790</td>
<td>Add changesets to the review should be inactive for closed or abandoned review</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-788</td>
<td>Add double click listener to Add Changesets Page</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-786</td>
<td>Improve handling of SCM connectors/bridges (missing when one installs connector from Discovery portal). At least some info message about how to install them would be appreciated. Currently it generated a lot of support cases.</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-785</td>
<td>Add usage reports for Eclipse versions and OSes</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-784</td>
<td>Add handling for dirty files when creating post-commit review (merging with create pre-commit action?)</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-783</td>
<td>Create post commit review from multiple file selection</td>
<td>Closed</td>
<td></td>
</tr>
<tr>
<td>PLE-781</td>
<td>Connection problems when removing and then adding the same server</td>
<td>Resolved</td>
<td></td>
</tr>
<tr>
<td>PLE-780</td>
<td>NPE in the plugin</td>
<td>Resolved</td>
<td></td>
</tr>
</tbody>
</table>
Atlassian Connector for Eclipse - v2.0 Upgrade Notes

Below are some essential notes on upgrading to version 2.0 of the Atlassian IDE Plugin for Eclipse. For details of the new features and improvements in this release, please read the Release Notes.

Upgrading from Mylyn JIRA Connector for Eclipse

If you are currently using the legacy Mylyn JIRA Connector (e.g. v3.3.1, v3.3.2, etc) with Eclipse, you will need to follow the instructions below to upgrade to the Atlassian Connector for Eclipse v2.0.

1. Add the Atlassian Connector for Eclipse update site to Eclipse, if you haven't done so already. For example, to do this in Eclipse 3.5.1 Galileo carry out the following steps:
   a. Click the 'Help' menu and select 'Install New Software'. The 'Install' window will display.
   b. Click the 'Add' button. Enter the appropriate site information, as described in Installing the Eclipse Connector.
2. Once you have added the update site, you will need to check for updates for the Mylyn JIRA Connector and update it to 3.4.0 stub. For example, to do this in Eclipse 3.5.1 Galileo carry out the following steps:
   a. Click the 'Help' menu and select 'Check for Updates'. The 'Available Updates' window will display.
   b. Tick the 'Mylyn Connector: JIRA (Compatibility)' checkbox (version should be '3.4.0') and click the 'Next Button'.
   c. If prompted, accept the license agreement and click the 'Finish' button.
   d. When prompted, restart Eclipse.
3. You can now install the Atlassian Connector for Eclipse, as described in Installing the Eclipse Connector.
Eclipse upgrade mechanism limitations (https://bugs.eclipse.org/bugs/show_bug.cgi?id=303260 and https://bugs.eclipse.org/bugs/show_bug.cgi?id=295675) mean that if you try to install the latest version of the Atlassian Connector for Eclipse over the top of an old Mylyn JIRA Connector installation, you will get an error similar to the following one:

Cannot complete the install because of a conflicting dependency.
Software being installed: Atlassian JIRA Connector (Recommended) 2.0.0.v20100224
(com.atlassian.connector.eclipse.jira.feature.group 2.0.0.v20100224)
Software currently installed: Mylyn Connector: JIRA 3.3.1.v20091215-0000-e3x
(org.eclipse.mylyn.jira_feature.feature.group 3.3.1.v20091215-0000-e3x)
Only one of the following can be installed at once:
Mylyn Connector: JIRA (Compatibility) 3.4.0 (org.eclipse.mylyn.jira_feature.feature.jar 3.4.0)
Mylyn Connector: JIRA 3.3.1.v20091215-0000-e3x (org.eclipse.mylyn.jira_feature.feature.jar 3.3.1.v20091215-0000-e3x)
Cannot satisfy dependency:
  From: Atlassian JIRA Connector (Recommended) 2.0.0.v20100224
  (com.atlassian.connector.eclipse.jira.feature.group 2.0.0.v20100224)
  To: org.eclipse.mylyn.jira_feature.feature.group [3.4.0]
Cannot satisfy dependency:
  From: Mylyn Connector: JIRA 3.3.1.v20091215-0000-e3x
  (org.eclipse.mylyn.jira_feature.feature.group 3.3.1.v20091215-0000-e3x)
  To: org.eclipse.mylyn.jira_feature.feature.jar [3.3.1.v20091215-0000-e3x]
Cannot satisfy dependency:
  From: Mylyn Connector: JIRA (Compatibility) 3.4.0
  (org.eclipse.mylyn.jira_feature.feature.jar 3.4.0)
  To: org.eclipse.mylyn.jira_feature.feature.jar [3.4.0]

We are working with Tasktop to resolve this problem.

RELATED TOPICS
Atlassian Connector for Eclipse - v2.0 Release Notes

**Atlassian Connector for Eclipse - v2.0 Beta Release Notes**

**24 February 2010**

With pleasure, Atlassian presents **version 2.0 beta** of the Atlassian Connector for Eclipse.

We've overhauled **Crucible** reviews in your Eclipse IDE this release. The new Crucible Review perspective allows you to work on the review as well as the file content at the same time, without having to switch back and forth between the two. The Crucible Review perspective also organises your workspace into a logical layout that makes working with reviews even easier.

You can now create post-commit Crucible reviews from multiple files within your Eclipse IDE, via the Package Explorer, or from a Java file viewed in the editor. Files selected in the Package Explorer can also be added to active reviews.

The connector now supports additional SCM systems for post-commit reviews. Post-commit reviews created from changesets now support all SCMs currently supported by Crucible.

There are a number of smaller improvements and bug fixes, listed below.

---

**Important Upgrade Notes**
The **Atlassian Connector for Eclipse - v2.0 Beta Upgrade Notes** contains important information that you must read before upgrading.

---

**Don’t have the Atlassian Connector for Eclipse yet?**

You can install the connector directly from the Eclipse software updates manager or from a zipped archive. Follow our installation guide.

---

**Highlights of this Release:**

- Crucible Review Perspective
- Improvements to Crucible Review Creation
- Addition of Files to Reviews
- Support for Additional SCM Systems for Post-Commit Reviews
- Other Things Worth Mentioning

⭐️ **We love your feedback!** ⭐️

Please log your issues, requests and votes. They help us decide what needs doing.
This is an open source project. You can get the source code from our SVN repository.

**Highlights of this Release**

1. **Crucible Review Perspective**

   The new Crucible Review perspective allows you to work on the review as well as the file content at the same time. You can navigate, view/add/remove comments and more; all without having to switch back and forth between the two. The Crucible Review perspective also organises your workspace into a logical layout that makes working with reviews even easier.

   The new Crucible Review perspective also presents you with a workspace layout that makes working with reviews even easier. Key information about the review is displayed in a new Review Explorer in this perspective. The Review Explorer also keeps all of the commonly used review functions at your fingertips. Finally, the new perspective also includes a new Review Comment view. This view provides you with the information and functions you need to deal with individual review comments, all in one place.

2. **Improvements to Crucible Review Creation**

   We've added a number of new ways for you to create reviews in your Eclipse IDE. Post-commit reviews can now be created from files selected in the Package Explorer or from the Java file you are currently viewing in the Active Task Editor. You can also add a file to an active review via the Package Explorer.
Addition of Files to Reviews

In addition to new ways to create reviews in your Eclipse IDE, you can now also add files to reviews.
Support for Additional SCM Systems for Post-Commit Reviews

The connector now supports all SCM systems supported by Crucible for post-commit reviews created from changesets. For information on the SCM systems supported by Crucible, i.e. Subversion, CVS, Perforce, Git and IBM Clearcase. Please see Crucible Supported Platforms for more information.

Other Things Worth Mentioning

This release brings a number of bug fixes and improvements. The following items are worth a special mention:

- Publish all draft comments with a single click.
- Create ad-hoc reviews directly from your code.
- New functions for working with comments — next/previous comment navigation, filter out read comments, mark comments as read/unread, reply to replies.
- Support for working on reviews without moderators (note, creation of reviews without moderators is not supported yet).
- Pre-defined filters in the New Query wizard for JIRA.
- Restrict JIRA comment visibility to specific roles.

Complete List of Fixes in this Release

<table>
<thead>
<tr>
<th>JIRA Issues (122 issues)</th>
<th>Key</th>
<th>Summary</th>
<th>Priority</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLE-1041</td>
<td></td>
<td>Can’t cancel mark as read job in Comment View after refresh</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JIRA ID</td>
<td>Description</td>
<td>Status</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------------------------------------------------------------</td>
<td>--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1039</td>
<td>Annotations not visible for local file opened from Review View if it was already open in the editor</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1038</td>
<td>[regression] problems in Review Perspective</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1037</td>
<td>Ability to refresh review from Review View</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1028</td>
<td>Creating review with comment from changed file (editor) does not add a comment</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1025</td>
<td>Comments added to the compare editor are not visible</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1024</td>
<td>Problem when opening file from Review Explorer</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1022</td>
<td>Review Explorer synchronize with editor problem</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1019</td>
<td>Delay marking the comment as read in Review Explorer / Comment View</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1018</td>
<td>Support Wiki markup when adding crucible comment</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1017</td>
<td>Unread comments filter should not hide comments</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1011</td>
<td>Review Explorer focus problem</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1008</td>
<td>Eclipse connector is unable to log in to latest Studio</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1003</td>
<td>Implement new usage reports on our wiki</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1001</td>
<td>jira query wizard does not allow to combine &quot;no component&quot; with other concrete components for 'in components' filter</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-1000</td>
<td>jira query wizard does not allow to combine &quot;no version&quot; with other concrete versions for 'reported in' filter</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-999</td>
<td>NPE in Create Review wizard (Back button does not work)</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-997</td>
<td>[regression] Add to Active review does not work from the editor - it wants to create a new review instead</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-993</td>
<td>Reviewers list broken in review details window (no display name)</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-990</td>
<td>People installing just JIRA Connector should get our Support (StAC) JIRA server added automatically as a Task Repository</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-988</td>
<td>[installation] Bamboo integration is installed when the user just selects Atlassian Connector for Eclipse (Required) feature</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-987</td>
<td>Show referenced file revisions for a single comment in Comment View</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-986</td>
<td>Support wiki markup in Crucible review comments and statement of objectives</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-984</td>
<td>Mylyn JIRA connector hyperlink detector doesn't always correctly detect JIRA issue hyperlinks</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-983</td>
<td>New comment part error when creating new task</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-982</td>
<td>Synchronize problems in review perspective</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-979</td>
<td>Double clicking in Review Explorer on a comment to changed XML file opens this file, but moves the</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-977</td>
<td>Sort line comments in Review Explorer tree by lines first, not just by dates</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-976</td>
<td>Tooltip for file nodes in Review Explorer</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-975</td>
<td>Link with editor setting in Review Explorer is not saved</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-974</td>
<td>SWTError in Review Perspective when linking is on</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-973</td>
<td>[regression] invalid compacting of review tree in cases when directories themselves are review items</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-972</td>
<td>Show revision string in file name tab (editor) for resources fetched externally while doing review</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-970</td>
<td>Atlassian Connector features (and logo) are not displayed after upgrading just Mylyn 3.3.x to Mylyn 3.3.99</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-968</td>
<td>Widget is disposed exception</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-966</td>
<td>[usage reporting] Add info about which Atlassian features are installed</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-952</td>
<td>Selection (and Comment View) is not correctly refreshed when a comment is changed (e.g. due to update or publishing)</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-946</td>
<td>Show reply in review explorer tree after it's has just been added by the user</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-944</td>
<td>Align reviewers list widget with objectives/comment text box in the last page of create review wizard.</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-943</td>
<td>Next/Prev comment action does not sync with Comment view</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-940</td>
<td>Tree view in Review Wizard shows multiple entries for the same path</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-939</td>
<td>Get repository changes directly from Crucible using new REST API</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-938</td>
<td>Separate 'objectives' and 'comment' text controls when creating review from selection</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-937</td>
<td>Select/unselect with spacebar does not work as expected.</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-936</td>
<td>Allow to create pre-commit review if there is no connector for selected files (for create review action)</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-935</td>
<td>Ability to add general comment from Review Explorer View</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-932</td>
<td>fisheye integration does network I/O in UI thread when right clicking resource</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-930</td>
<td>[regression] 3.3.0 breaks Update Attributes From Repository on JIRA 3.6 and earlier</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-929</td>
<td>investigate possible race condition in JiraRepositoryConnectorTest.testMarkStaleClosedTask</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-925</td>
<td>provide UI for setting comment visibility</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-913</td>
<td>Green Hopper ranking custom field is erased on submit</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-908</td>
<td>Query with &quot;assigned to&quot; field set to unassigned is ignored</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-907</td>
<td>when editing jira task, it removes attributes from the issue, specifically the &quot;Fix Version/s&quot; field</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JIRA Key</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-905</td>
<td>jira query wizard don't allow to combine &quot;fix for&quot; &quot;no fix version&quot; and other concrete versions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-891</td>
<td>reassigning an issue and stopping progress in the same 'submit' causes undesirable results</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-874</td>
<td>Do not allow posting non-draft replies to drafts</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-873</td>
<td>Move general comments to Review Explorer view</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-872</td>
<td>Opening a file or a comment from Review Explorer should automatically show Review Comments view</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-871</td>
<td>Merge Create Review from selected resource(s) and Discuss code actions into one action</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-870</td>
<td>Display all Crucible servers in Task Repository selection wizard page and disable those which are not capable of handling given request</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-869</td>
<td>Add a tool-tip for pre-commit entries in Create Review wizard explaining what pre-commit means here</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-868</td>
<td>Remove &quot;post-commit&quot; decoration from Create Review wizard</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-867</td>
<td>Enhance error handling</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-864</td>
<td>Widget is disposed bug</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-862</td>
<td>NPE when playing with Crucible perspective</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-860</td>
<td>Open remote files using Crucible 2.x API</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-859</td>
<td>Wrong url used to validate mappings for Subversion</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-856</td>
<td>Display general comments in Review Explorer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-855</td>
<td>Use icon decorations for modified/added/removed state in Review Explorer instead of decoration</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-854</td>
<td>Navigate next/prev comment in Review Explorer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-851</td>
<td>use colours in Review Explorer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-850</td>
<td>Comments marked as defects are not marked in Review Explorer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-849</td>
<td>Synchronize Crucible Explorer View and Crucible Comments View with Editor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-848</td>
<td>Add show only unread comments filter to Review Explorer View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-847</td>
<td>Use images in Crucible Explorer View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-846</td>
<td>Compact entries in Crucible Explorer View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-845</td>
<td>Handle double-click in Crucible Explorer View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-844</td>
<td>Add Expand All / Collapse All toolbar actions in Crucible Explorer View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-842</td>
<td>Add shortcut icon for review editor in Explorer and Comments views</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-841</td>
<td>Having wrong mapping gives me HTTP400 from Crucible</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-840</td>
<td>Facilitating installation of Connector when only JIRA Connector (listed on Mylyn Discovery) is installed</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-839</td>
<td>Empty error message in Error Log view</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-837</td>
<td>Show test results action doesn't compile on e3.6</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-834</td>
<td>Analyse inconsistency in recent usage reports and fix them</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-807</td>
<td>Improve explanation why Repository mappings are needed and how to set them up</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-806</td>
<td>Do not automatically propose a mapping (add mapping dialog) when accessing the mapping dialog via Define Repository Mappings from Select Changeset wizard page when all required mappings are present</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-804</td>
<td>Merge Create Pre-commit and Post-commit Review actions into one action which is capable of handling mixed reviews (with post-commit items and pre-commit items together)</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-803</td>
<td>Do not ask user whether to open Bamboo view (after Bamboo task repo is added) and just focus on this view.</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-802</td>
<td>Support dirty files in &quot;Add files to active review&quot; - add them in pre-commit mode</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-801</td>
<td>In create review from changeset page: display which changeset (from which SCM) does not have correct mapping configured</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-799</td>
<td>Add separator in Team menu between FishEye and Crucible related actions</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-797</td>
<td>Support reviews without the moderator (introduced with Crucible 2.2M1)</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-796</td>
<td>Use Crucible/FishEye mappings when opening files from Crucible Review</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-793</td>
<td>Cannot create post-commit review if selection contains svn-ignored file</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-791</td>
<td>Duplicated entries in pre-commit review</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-790</td>
<td>Add changesets to the review should be inactive for closed or abandoned review</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-788</td>
<td>Add double click listener to Add Changesets Page</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-786</td>
<td>Improve handling of SCM connectors/bridges (missing when one installs connector from Discovery portal). At least some info message about how to install them would be appreciated. Currently it generated a lot of support cases.</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-785</td>
<td>Add usage reports for Eclipse versions and OSes</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-784</td>
<td>Add handling for dirty files when creating post-commit review (merging with create pre-commit action?)</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-783</td>
<td>Create post commit review from multiple file selection</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-781</td>
<td>Connection problems when removing and then adding the same server</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-780</td>
<td>NPE in the plugin</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-773</td>
<td>Read/Unread support for comments</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-772</td>
<td>Reporting of new opt-in opt-out metrics</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>----------------------------------------</td>
<td>--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-766</td>
<td>Automatically add commented file to active review when this file is not in review scope</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-765</td>
<td>Ability to add selected files(s) and or package(s) to active review</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-761</td>
<td>Add most common predefined filters to New Query wizard for JIRA Connector</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-757</td>
<td>Bamboo view shows Initializing view... until refreshed</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-744</td>
<td>New (view based) approach to navigate through Crucible reviews (was: adding or removing a review comment crashes Eclipse 3.6M2 on GIIK)</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-740</td>
<td>Add ability to create a review from selected lines in a text editor</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-738</td>
<td>Add ability to create a review from a package or directory</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-720</td>
<td>Create perspective for reviewing</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-712</td>
<td>show tests results fails with NoSuchMethodError on 3.6M2</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-710</td>
<td>Having two reviews with the same file causes a problem</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-693</td>
<td>Show comment visibility in task editor if applicable</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-661</td>
<td>JIRA Comment Visibility (roles &amp; groups)</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-455</td>
<td>Marking a review as complete does not give you an option to post all drafts</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-442</td>
<td>Post all draft Action</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-396</td>
<td>A user should be able to create a new post commit review from a file opened in the java editor</td>
<td>Closed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-339</td>
<td>Open the Bamboo plans view after configuration</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLE-310</td>
<td>Review Annotations for non Java files (i.e. .xml, .mf)</td>
<td>Resolved</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Atlassian Connector for Eclipse - v2.0 Beta Upgrade Notes**

Below are some essential notes on upgrading to version 2.0 Beta of the Atlassian IDE Plugin for Eclipse. For details of the new features and improvements in this release, please read the Release Notes.

**Upgrading from Mylyn JIRA Connector for Eclipse**

If you are currently using the legacy Mylyn JIRA Connector (e.g. v3.3.1, v3.3.2, etc) with Eclipse, you will need to follow the instructions below to upgrade to the Atlassian Connector for Eclipse v2.0 Beta.

1. Add the Atlassian Connector for Eclipse update site to Eclipse, if you haven't done so already. For example, to do this in Eclipse 3.5.1 Galileo carry out the following steps:
   a. Click the 'Help' menu and select 'Install New Software'. The 'Install' window will display.
   b. Click the 'Add' button. Enter the appropriate site information, as described in Installing the Eclipse Connector.
2. Once you have added the update site, you will need to check for updates for the Mylyn JIRA Connector and update it to 3.4.0 stub. For example, to do this in Eclipse 3.5.1 Galileo carry out the following steps:
   a. Click the 'Help' menu and select 'Check for Updates'. The 'Available Updates' window will display.
   b. Tick the 'Mylyn Connector: JIRA (Compatibility)' checkbox (version should be '3.4.0') and click the 'Next Button'.
   c. If prompted, accept the license agreement and click the 'Finish' button.
   d. When prompted, restart Eclipse.
3. You can now install the Atlassian Connector for Eclipse, as described in Installing the Eclipse Connector.

**Background Information**

Eclipse upgrade mechanism limitations (https://bugs.eclipse.org/bugs/show_bug.cgi?id=303260 and
https://bugs.eclipse.org/bugs/show_bug.cgi?id=295675] mean that if you try to install the latest version of the Atlassian Connector for Eclipse over the top of an old Mylyn JIRA Connector installation, you will get the following error:

```
Cannot complete the install because of a conflicting dependency.
Software being installed: Atlassian JIRA Connector (Recommended) 2.0.0.v20100224-BETA
(com.atlassian.connector.eclipse.jira.feature.group 2.0.0.v20100224-BETA)

Software currently installed: Mylyn Connector: JIRA 3.3.1.v20091215-0000-e3x
(org.eclipse.mylyn.jira_feature.feature.group 3.3.1.v20091215-0000-e3x)

Only one of the following can be installed at once:
Mylyn Connector: JIRA (Compatibility) 3.4.0 (org.eclipse.mylyn.jira_feature.feature.jar 3.4.0)
Mylyn Connector: JIRA 3.3.1.v20091215-0000-e3x (org.eclipse.mylyn.jira_feature.feature.jar
3.3.1.v20091215-0000-e3x)

Cannot satisfy dependency:
  From: Atlassian JIRA Connector (Recommended) 2.0.0.v20100224-BETA
  To: org.eclipse.mylyn.jira_feature.feature.group [3.4.0]

Cannot satisfy dependency:
  From: Mylyn Connector: JIRA 3.3.1.v20091215-0000-e3x
  To: org.eclipse.mylyn.jira_feature.feature.jar [3.3.1.v20091215-0000-e3x]

Cannot satisfy dependency:
  From: Mylyn Connector: JIRA (Compatibility) 3.4.0
  To: org.eclipse.mylyn.jira_feature.feature.group [3.4.0]
  To: org.eclipse.mylyn.jira_feature.feature.jar [3.4.0]
```

We are working with Tasktop to resolve this problem.

RELATED TOPICS

Atlassian Connector for Eclipse - v2.0 Beta Release Notes

Atlassian Connector for Eclipse - v1.3.1 Release Notes

13 November 2009

With pleasure, Atlassian presents version 1.3.1 of the Atlassian Connector for Eclipse.

This is primarily a bug-fix release. Due to a bug in Bamboo 2.3.x REST API, the connector did not work with Bamboo 2.3. This issue is now fixed. Note that we have reverted to using the old remote API for this Bamboo version, so the connector will not show builds in progress for Bamboo 2.3.

The connector now also displays correctly all comments from iterative reviews managed with Crucible 2.1, which uses a slightly different and more flexible way to serve information about them.

Below is the complete list of fixes and improvements in this release.

### Don't have the Atlassian Connector for Eclipse yet?

You can install the connector directly from the Eclipse software updates manager or from a zipped archive. Follow our installation guide.

🌟 We love your feedback! 🌟

Please log your issues, requests and votes. They help us decide what needs doing. This is an open source project. You can get the source code from our SVN repository.

Complete List of Fixes in this Release

<table>
<thead>
<tr>
<th>JIRA Issues (4 issues)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
</tr>
<tr>
<td>PLE-758</td>
</tr>
<tr>
<td>PLE-755</td>
</tr>
<tr>
<td>PLE-754</td>
</tr>
<tr>
<td>PLE-748</td>
</tr>
</tbody>
</table>

Atlassian Connector for Eclipse - v1.3 Release Notes

5 November 2009
With pleasure, Atlassian presents version 1.3 of the Atlassian Connector for Eclipse.

You can now perform pre-commit Crucible reviews from within your Eclipse IDE, as well as the post-commit reviews that were already available. Crucible's web interface lets you do reviews on patches, but you must make and upload the patch. Now it is easier and simpler to do a pre-commit review from within Eclipse.

The connector's Bamboo component now displays builds that are in progress or in the queue, as well as the completed builds.

We are delighted to announce that the connector's JIRA component is now much faster. The team has been working with Tasktop on a number of improvements, now available with the latest Eclipse Mylyn release 3.3. In particular, you will notice a marked reduction in the time it takes to load your JIRA issues into the IDE.

There are a number of smaller improvements and bug fixes, listed below. Please note also that, effective from this release, the connector no longer supports Eclipse 3.3 (Europa).

Don't have the Atlassian Connector for Eclipse yet?

You can install the connector directly from the Eclipse software updates manager or from a zipped archive. Follow our installation guide.

Highlights of this Release:

- Crucible Pre-Commit Reviews
- Bamboo Builds in Progress or in Queue
- JIRA Much Faster
- Other Things Worth Mentioning

🌟 We love your feedback! 🌟

Please log your issues, requests and votes. They help us decide what needs doing.

This is an open source project. You can get the source code from our SVN repository.

Using Bamboo 2.3?

Unfortunately due to a bug with the REST API, this release of the Atlassian Connector for Eclipse will not work with Bamboo 2.3. We are working on a fix for this. As a workaround, please consider upgrading to the latest Bamboo release.

Highlights of this Release

1

Crucible Pre-Commit Reviews

Would you like your team members to review your code before you commit it to the repository? Although Crucible's web interface lets you do reviews on patches, you need to make and upload the patch before you can do the review. Now you can use the connector to do a pre-commit review from within Eclipse. The review gathers your local working copies of the files and makes them available to all reviewers for comment.
You and your reviewers can then open the changed files in Eclipse and add review comments, even though the changes have not yet been committed to the source repository. (Note: Currently the connector supports pre-commit reviews with Subclipse and Subversive only. You cannot yet do pre-commit reviews using CVS.)
Bamboo Builds in Progress or in Queue

The connector now displays builds that are in progress or in the queue, as well as all completed builds. Note that you need Bamboo 2.4 or later to make use of this feature.

JIRA Much Faster

Thanks to lazy and smarter fetching and generally improved RPC handling, the connector’s JIRA component is now much faster. Now available with the latest Eclipse Mylyn download, the efficiency and other improvements are the result of our work with the Tasktop team. See the Mylyn release notes for Mylyn version 3.3. Here is a summary:

- **Improved performance**: The connector now employs lazy loading, so that it fetches only the data it requires instead of loading all JIRA issue metadata at once. You will notice a marked improvement in the loading time when:
  - Adding a JIRA task repository,
  - Creating a JIRA issue filter,
  - Adding a JIRA issue.
- **Progress monitor**: There is now a better indication of progress when adding a JIRA repository.
- **Filtering of version and component**: You can now select versions and components from a filtered list.

Other Things Worth Mentioning

This release brings a number of bug fixes and improvements. The following items are worth a special mention:

- When you create a new task from a failed Bamboo build, the task will include links to the relevant JIRA issues.
- In the Bamboo build window, the related JIRA issue keys are now hyperlinked so that you can click through to the issue.
- We have fixed a number of bugs in Crucible reviews and improved the related error messages to give more information about the problem encountered.
- The Crucible component now includes review navigation, allowing you to navigate between comments and files in the review.
- Eclipse 3.3 (Europa) is no longer supported. Please note that, effective from version 1.3 of the Atlassian Connector for Eclipse, the connector will no longer support Eclipse 3.3 (Europa). We recommend that you upgrade your Eclipse platform to get the best benefit from Eclipse Mylyn and the Atlassian Connector for Eclipse.

Complete List of Fixes in this Release

<table>
<thead>
<tr>
<th>JIRA Issues (55 issues)</th>
<th>Priority</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLE-743</td>
<td></td>
<td></td>
</tr>
<tr>
<td>installed plugin appears to block update of Mylyn to 3.3.0.v20091015-0500-e3x</td>
<td>![Symbol]</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-735</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cannot open review files</td>
<td>![Symbol]</td>
<td></td>
</tr>
<tr>
<td>Issue</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>PLE-734</td>
<td>Plugin should remember last selected SCM provider when creating pre-commit review</td>
<td></td>
</tr>
<tr>
<td>PLE-731</td>
<td>Direct Click Through should not show a dialog if it fails to bind to the port</td>
<td></td>
</tr>
<tr>
<td>PLE-729</td>
<td>Problem when opening binary files for post commit review</td>
<td></td>
</tr>
<tr>
<td>PLE-727</td>
<td>Outdated annotation displayed in the editor</td>
<td></td>
</tr>
<tr>
<td>PLE-726</td>
<td>Cannot install onto latest SpringSource Tool Suite (2.2.0)</td>
<td></td>
</tr>
<tr>
<td>PLE-725</td>
<td>Clicking Favourites in Bamboo Task Repository does nothing</td>
<td></td>
</tr>
<tr>
<td>PLE-718</td>
<td>Remember what review type user usually creates</td>
<td></td>
</tr>
<tr>
<td>PLE-716</td>
<td>Direct Click Through on Linux-Gtk creates multiple gray windows</td>
<td></td>
</tr>
<tr>
<td>PLE-714</td>
<td>Pictures (and probably other binary files) are displayed in text editor (as of course rubbish) when clicked on file hyperlink in review editor for pre-commit review</td>
<td></td>
</tr>
<tr>
<td>PLE-707</td>
<td>Show Repositories View from Bamboo View toolbar shows Task List view instead</td>
<td></td>
</tr>
<tr>
<td>PLE-706</td>
<td>I get strange message when opening editor for annotated ADDED file</td>
<td></td>
</tr>
<tr>
<td>PLE-704</td>
<td>Error removing file comment (line or general) which has associated reply</td>
<td></td>
</tr>
<tr>
<td>PLE-702</td>
<td>Disconnecting Bamboo Task Repository does not remove its plans from Bamboo View</td>
<td></td>
</tr>
<tr>
<td>PLE-701</td>
<td>When a build is in progress while auto-refresh of Bamboo View, I get notification suggesting that the plan is disabled</td>
<td></td>
</tr>
<tr>
<td>PLE-699</td>
<td>NPE when refreshing Bamboo View and empty plan is selected</td>
<td></td>
</tr>
<tr>
<td>PLE-696</td>
<td>When build is in progress (throbber icon in State column is displayed) then I cannot actually scroll my Bamboo View</td>
<td></td>
</tr>
<tr>
<td>PLE-691</td>
<td>Drop eclipse 3.3 support</td>
<td></td>
</tr>
<tr>
<td>PLE-689</td>
<td>Provide a meaningful message when Crucible server has Remote API turned off</td>
<td></td>
</tr>
<tr>
<td>PLE-687</td>
<td>Building a release should be a one click operation (was: fix build system)</td>
<td></td>
</tr>
<tr>
<td>PLE-686</td>
<td>bug #255430 - more informative progress monitor when adding repository</td>
<td></td>
</tr>
<tr>
<td>PLE-685</td>
<td>“update task attributes” button only fetches data for single project</td>
<td></td>
</tr>
<tr>
<td>PLE-679</td>
<td>lots of Editor is not an ITextEditor or editor inputs... errors in log</td>
<td></td>
</tr>
<tr>
<td>PLE-676</td>
<td>Update Repository Data does not work for recently changed (added/removed) Crucible projects</td>
<td></td>
</tr>
<tr>
<td>PLE-675</td>
<td>Confusing message when opening editor for reviewed file and there is no active issue</td>
<td></td>
</tr>
<tr>
<td>PLE-665</td>
<td>Better support for assign issue operation: should respect assign issue permission rather than edit issue (used now)</td>
<td></td>
</tr>
<tr>
<td>PLE-658</td>
<td>Ability to create a pre-commit review from local files (for Subversive)</td>
<td></td>
</tr>
<tr>
<td>JIRA ID</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>PLE-656</td>
<td>JIRA task editor should not update the task if only comment changed</td>
<td></td>
</tr>
<tr>
<td>PLE-651</td>
<td>Fix usage reports on EAC</td>
<td></td>
</tr>
<tr>
<td>PLE-650</td>
<td>Edit Repository Query - filter list should grab horizontal space</td>
<td></td>
</tr>
<tr>
<td>PLE-649</td>
<td>Reverse version list (present them the same order as JIRA does)</td>
<td></td>
</tr>
<tr>
<td>PLE-648</td>
<td>JIRA issue editor should show versions as a comma separated value list</td>
<td></td>
</tr>
<tr>
<td>PLE-646</td>
<td>Restore build selection after refresh in build view</td>
<td></td>
</tr>
<tr>
<td>PLE-639</td>
<td>[Collection] JIRA Bugs for M7</td>
<td></td>
</tr>
<tr>
<td>PLE-635</td>
<td>JIRA Connector should set User-Agent HTTP header to something more meaningful and consistent</td>
<td></td>
</tr>
<tr>
<td>PLE-616</td>
<td>Indicate a build is in progress</td>
<td></td>
</tr>
<tr>
<td>PLE-615</td>
<td>Ability to work on a pre-commit review created from Eclipse</td>
<td></td>
</tr>
<tr>
<td>PLE-614</td>
<td>Ability to create a pre-commit review from local files (for Subclipse)</td>
<td></td>
</tr>
<tr>
<td>PLE-610</td>
<td>Direct Click Through support for Bamboo</td>
<td></td>
</tr>
<tr>
<td>PLE-580</td>
<td>Default selection of Crucible Project in New Review wizard is remembered per-repository</td>
<td></td>
</tr>
<tr>
<td>PLE-570</td>
<td>confusing errors when creating a review with incomplete Subclipse install</td>
<td></td>
</tr>
<tr>
<td>PLE-511</td>
<td>validation error not very helpful</td>
<td></td>
</tr>
<tr>
<td>PLE-508</td>
<td>combo boxes are empty when no fisheye repositories are defined</td>
<td></td>
</tr>
<tr>
<td>PLE-493</td>
<td>Review activation settings do not work per-server but rather globally</td>
<td></td>
</tr>
<tr>
<td>PLE-491</td>
<td>Initial metadata fetch is very slow for JIRA</td>
<td></td>
</tr>
<tr>
<td>PLE-461</td>
<td>Incomplete scroll bars appearing in CRU Editor</td>
<td></td>
</tr>
<tr>
<td>PLE-457</td>
<td>Make SVN-cruible repository mappings a repositorySetting</td>
<td></td>
</tr>
<tr>
<td>PLE-445</td>
<td>If review creation fails, wizard should stay open to allow user to react</td>
<td></td>
</tr>
<tr>
<td>PLE-423</td>
<td>[refactoring] Bucket for refactoring work that needs to or should be done</td>
<td></td>
</tr>
<tr>
<td>PLE-418</td>
<td>Emptiness under Statement of Objectives in review editor</td>
<td></td>
</tr>
<tr>
<td>PLE-417</td>
<td>Poor error handling while refreshing Crucible editor</td>
<td></td>
</tr>
<tr>
<td>PLE-345</td>
<td>Add comment dialog does not use dialog font</td>
<td></td>
</tr>
<tr>
<td>PLE-258</td>
<td>Bamboo UI Tweaks</td>
<td></td>
</tr>
</tbody>
</table>
Atlassian Connector for Eclipse - v1.2 Release Notes

19 August 2009

With pleasure, Atlassian presents version 1.2 of the Atlassian Connector for Eclipse.

It's now easier than ever to install the Atlassian Connector for Eclipse. For those who cannot use our update site (for example, those who are working behind a firewall) or just prefer local installation, we now offer a zipped version of the entire update site. There is also a weekly update site available. Refer to our installation guide for details.

This release introduces the collection of usage statistics to help us better understand how people use the connector, so that we can work on improving the user experience. When you first open your IDE with the connector installed, we ask whether you agree to participate in the collection of usage statistics. You can also change your decision later via the configuration settings, as described in our configuration guide.

Not sure how to use the Crucible, FishEye or Bamboo components of the connector, or want to see what they can do for you? We now have detailed user guides:

- Working with Crucible Reviews in Eclipse
- Working with your FishEye Repository View in Eclipse
- Working with Bamboo Builds in Eclipse

Don't have the Atlassian Connector for Eclipse yet?

You can install the connector directly from the Eclipse software updates manager, via the Mylyn Connector Discovery wizard or from a zipped archive. Follow our installation guide.

This is an open source project. The source code is available in our SVN repository.

🌟 We love your feedback! 🌟

Please log your issues, requests and votes. They help us decide what needs doing.

Complete List of Fixes in this Release

<table>
<thead>
<tr>
<th>JIRA Issues (20 issues)</th>
<th>Priority</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLE-637 Query Title should be automatically filled out for predefined queries in New Query dialog</td>
<td>🟢</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-632 two identical usage reports in New wizard</td>
<td>🟢</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-630 Textual changes on Usage Tracking screens</td>
<td>🟢</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-628 Fix help link on “Usage Data” preferences window</td>
<td>🟢</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-626 Open the Task Repository view opens Task List instead</td>
<td>🟢</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-617 ensure forked version of mylyn usage plugin does not conflict with the released mylyn usage monitor</td>
<td>🟢</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-613 Create a weekly update site</td>
<td>🟢</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-598 Create Upload Server</td>
<td>🟢</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-590 Allow user to get to Eclipse from CRU, FE, and JIRA.</td>
<td>🟢</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-589 Error trying to open file from Crucible Code Review Editor</td>
<td>🟢</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-588 New Review wizard fails to get Crucible repos when Save Password is disabled, but correct password was provided when prompted</td>
<td>🟢</td>
<td>Resolved</td>
</tr>
</tbody>
</table>
Atlassian Connector for Eclipse - v1.1 Release Notes

24 June 2009

With pleasure, Atlassian presents version 1.1 of the Atlassian Connector for Eclipse.

The Atlassian Connector for Eclipse builds on Mylyn’s task-focused interface to integrate your Eclipse IDE closely with your JIRA issues, Crucible code reviews, FishEye repository views and Bamboo builds. For a full list of the connector's features, take a look at our version 1.0 release notes.

With the release of version 1.1, the Atlassian Connector for Eclipse is now fully compatible with Eclipse 3.5 (Galileo) and Mylyn 3.2.

This release introduces full support for Subversive where you are versioning your project under SVN. This means that you can work with issues, code reviews, builds and FishEye repository inspection when using the Subversive plugin for Eclipse, as well as the already-supported Subclipse plugin. Are you using Subclipse? The connector’s FishEye integration now provides full support for CVS.

You can now log work against a JIRA issue from within your Eclipse IDE. Combined with the connector's existing functionality, this means that you hardly ever need to move out of your IDE to your web browser when managing issues. We have also fixed a number of problems and concentrated on code refactoring.

We are changing the connector’s name, from ‘Atlassian Eclipse Connector’ to ‘Atlassian Connector for Eclipse’. The new name complies with the Eclipse Foundation’s Guidelines for Eclipse Logos & Trademarks.

Highlights of this Release:

- Highlights of this Release
  - Log Work against a JIRA Issue
  - Full Support for Subversive Plugin
  - FishEye Support for CVS
  - Other Points Worth Mentioning
  - Complete List of Fixes in this Release

Don’t have the Atlassian Connector for Eclipse yet?

You can install the connector directly from the Eclipse software updates manager or via the Mylyn Connector Discovery wizard. Follow our installation guide.

This is an open source project. The source code is available in our SVN repository.

🌟 We love your feedback! 🌟

Please log your issues, requests and votes. They help us decide what needs doing.
Note: Install either Subclipse or Subversive integration option, not both.

When installing components of the Atlassian Connector for Eclipse, please select either Subclipse or Subversive, not both. (Details are in our installation guide.) Installing one or the other of these two integration options will allow you to make full use of the Atlassian Connector's integration with Crucible for code reviews and FishEye for source repository viewing, when versioning your project under SVN.

Highlights of this Release

1

Log Work against a JIRA Issue

You can now log work against a JIRA issue from within your Eclipse IDE. Combined with the connector's existing functionality, this means that you hardly ever need to move out of your IDE to your web browser when managing issues.
Full Support for Subversive Plugin

The connector now provides full support for the Subversive plugin for Eclipse where you are versioning your project under SVN. Earlier releases of the connector already support Subclipse. This means that you can work with all the connector's Atlassian integrations from within your Eclipse IDE via Subversive or Subclipse.

- Manage JIRA issues.
- Conduct Crucible code reviews.
- Monitor Bamboo builds.
- Use FishEye's awesome repository inspection.

If you would like more details, please refer to our SCM support guide.

FishEye Support for CVS

The connector's FishEye integration now provides full support for CVS within Eclipse.

- Open a source file in FishEye's web interface, directly from your Eclipse editor.
- Copy the FishEye link to your clipboard and share FishEye file references with others.

If you would like more details, please refer to our SCM support guide.

Other Points Worth Mentioning

- When you are configuring a FishEye SCM mapping, the connector now provides a default SCM path.
- The connector supports Eclipse 3.5 (Galileo), which is due for GA (general availability) release on 24 June.
- The connector supports the new Mylyn 3.2 release, including a more streamlined and flexible bug reporting feature.
- You can now install the connector via Mylyn's new 'Connector Discovery' wizard, as described in our installation guide.
- In the Eclipse software installation dialogue, both Subversive and Subclipse are included in the 'optional' integration features rather than the 'recommended' features for the Atlassian connector.

Complete List of Fixes in this Release

Please take a look at the fixes in the connector's JIRA integration project on the Eclipse bug list.

Below is a list of the fixes and improvements in the main project for the Atlassian Connector for Eclipse. The list below is drawn from our JIRA issue tracker.

<table>
<thead>
<tr>
<th>JIRA Issues (42 issues)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>Summary</td>
<td>Priority</td>
</tr>
<tr>
<td>PLE-568</td>
<td>[subversive] create review from changeset fails</td>
<td></td>
</tr>
<tr>
<td>PLE-567</td>
<td>[subversive] compare editor viewer order mixed up</td>
<td></td>
</tr>
<tr>
<td>PLE-564</td>
<td>help points to non-existing wiki page</td>
<td></td>
</tr>
<tr>
<td>PLE-560</td>
<td>SCM problems</td>
<td></td>
</tr>
<tr>
<td>PLE-559</td>
<td>prepare better welcome screen messages</td>
<td></td>
</tr>
<tr>
<td>PLE-556</td>
<td>Please review the release notes for the Eclipse Connector 1.1</td>
<td></td>
</tr>
<tr>
<td>PLE-552</td>
<td>Documentation for Eclipse Connector 1.1</td>
<td></td>
</tr>
<tr>
<td>PLE-549</td>
<td>Update error messages to include Subversive support too</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-547</td>
<td>Subclipse shouldn't be installed if I don't select it</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-546</td>
<td>Consume name change for connector</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-544</td>
<td>Add contributions to Welcome page</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-541</td>
<td>Add ability to create Crucible review for Subversive changeset</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-539</td>
<td>Add Eclipse Connector section to Welcome (Overview) page</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-538</td>
<td>Auto populate SCM Path while configuring FishEye mapping for unmapped SCM</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-537</td>
<td>FishEye Subversive integration connects to repository when opening popup menu</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-536</td>
<td>Cannot see reviewers in Crucible Editor</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-534</td>
<td>[e3.5] team.main contribution is invalid under e3.5</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-533</td>
<td>Bamboo feature doesn't show up in Eclipse 3.5 during initial installation (but it does show up after calculating requirements)</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-532</td>
<td>Subversive support</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-531</td>
<td>Handle null returns in TeamUiUtils and ITeamResourceConnector</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-529</td>
<td>Eclipse Connector should not depend on Subversive and Subclipse</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-528</td>
<td>Empty implementations of CVS and Subversive Team Connectors break existing functionality</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-527</td>
<td>Build PLE depending on a concrete ACC version</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-526</td>
<td>NPE when opening FishEye link on CVS-managed project</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-524</td>
<td>Improved Subversive Support</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-522</td>
<td>FishEye vs Subversive integration</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-521</td>
<td>Open with FishEye for binary files should point to a FishEye history page (not raw file)</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-519</td>
<td>Switch to Subclipse 1.6</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-518</td>
<td>Add wizard banner to FishEye configuration mapping add/edit dialog</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-514</td>
<td>Crucible validation does not work</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-509</td>
<td>Support Open with FishEye for CVS projects</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-507</td>
<td>Select SCM repository dialog comes up empty</td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-503</td>
<td>“Get 10 more revisions” does not get 10 on Linux</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-502</td>
<td>implement support extension</td>
<td><img src="image" alt="Closed" /></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>PLE-473</td>
<td>[3.5] Issues with PLE and e3.5 Galileo</td>
<td><img src="image" alt="Closed" /></td>
</tr>
<tr>
<td>PLE-468</td>
<td>When opening a non-Java file for a review and annotations cannot be displayed, show a popup to advise the user</td>
<td><img src="image" alt="Closed" /></td>
</tr>
<tr>
<td>PLE-459</td>
<td>Provide ability to log work against JIRA task</td>
<td><img src="image" alt="Closed" /></td>
</tr>
<tr>
<td>PLE-452</td>
<td>[Collection] JIRA Bugs for M4</td>
<td><img src="image" alt="Closed" /></td>
</tr>
<tr>
<td>PLE-450</td>
<td>update build for Mylyn 3.2</td>
<td><img src="image" alt="Closed" /></td>
</tr>
<tr>
<td>PLE-427</td>
<td>CrucibleHttpSessionCallback and BambooHttpSessionCallback look suprisingly similar (time to merge them?)</td>
<td><img src="image" alt="Closed" /></td>
</tr>
<tr>
<td>PLE-305</td>
<td>[e3.5] annotation hovers are not displayed</td>
<td><img src="image" alt="Closed" /></td>
</tr>
<tr>
<td>PLE-295</td>
<td>[e3.5] key-conflict error in log</td>
<td><img src="image" alt="Closed" /></td>
</tr>
</tbody>
</table>

**Atlassian Eclipse Connector 1.0 Release Notes**

27 May 2009

With pleasure, Atlassian presents the **Atlassian Eclipse Connector 1.0**.

The Atlassian Eclipse Connector builds on Mylyn’s task-focused interface to integrate your Eclipse IDE closely with your JIRA issues, Crucible code reviews, FishEye repository views and Bamboo builds. With the earlier beta release we announced the improved connector resulting from the collaboration between Atlassian and Tasktop. Now we are delighted to present the production-ready 1.0 release.

What's new since the beta release?

- Use the new integration with Atlassian FishEye to open a source file in FishEye’s web interface directly from Eclipse, or to copy a FishEye URL to your clipboard and share it with others.
- Conduct your Crucible code reviews when versioning your project under SVN with Subversive or under CVS. Of course, the existing integration with Subclipse is also still available.
- Create a Crucible review from within Eclipse, for uncommitted code (‘patch’) or committed code (‘changeset’).
- Publish a draft review comment.
- Edit and remove existing comments.
- Assign reviewers without leaving Eclipse.
- Choose your favourite Bamboo builds as defined on the Bamboo server.

In the JIRA and Bamboo integrations we have focused on refactoring and bug-fixing, based on the results of testing the beta version.

**Highlights of this Release:**

- **FishEye Code Insight via Eclipse**
  - Configure your FishEye Server as a Mylyn Task Repository
  - Open a File in FishEye directly from Eclipse
- **Crucible Reviews in Eclipse**
  - Manage your Code Reviews via Mylyn’s Task-Focused Interface
  - Create a Crucible Review from within Eclipse
  - Conduct your Code Reviews in your IDE
  - See the Review Comments in Your Code
  - Work on the Review in the Diff View Too
  - Receive Notification of Review Updates
- **Bamboo Builds in Eclipse**
  - Configure your Bamboo Server as a Mylyn Task Repository
  - Manage your Bamboo Builds in your IDE
  - View the Build Logs
  - View the Changed Files
  - Monitor the Failed Tests
  - Receive Notification of Build changes
- **JIRA Issues in Eclipse**
  - Use the Awesome JIRA Mylyn Plugin as Part of the New Connector
  - Complete List of Fixes in this Release

**Don’t have the Atlassian Eclipse Connector yet?**

You can install the connector directly from the Eclipse software updates manager. Follow our [installation guide](image).
This is an open source project. The source code is available in our SVN repository.

🌟 We love your feedback! 🌟

Please log your issues, requests and votes. They help us decide what needs doing. It is easy to add an issue. If you do not yet have an account, the system will prompt you to create one.

### Upgrading from a previous version of the IDE Plugin for Eclipse

If you have previously installed version 0.4 or earlier of the Atlassian IDE Plugin for Eclipse, please uninstall the earlier version. Then follow the installation guide to install the new Atlassian Eclipse Connector.

### FishEye Code Insight via Eclipse

Even if you do not have Crucible, you can now hook up your FishEye server as an Eclipse task repository.

1. Configure your FishEye Server as a Mylyn Task Repository
   
   - Configure your FishEye server as a task repository.
   - Map your source repository to your FishEye repository.

2. Open a File in FishEye directly from Eclipse
   
   - Open a source file in FishEye's web interface, directly from your Eclipse editor.
   - Copy the FishEye link to your clipboard and share FishEye file references with others.
Enjoy Crucible's lightweight workflow to conduct code reviews within your IDE. Using the connector, you can handle your Crucible reviews as Eclipse Mylyn tasks and take advantage of Mylyn's powerful context management features. In addition, you can perform much of the review workflow without leaving your IDE.

### Manage your Code Reviews via Mylyn’s Task-Focused Interface

- Set up your Crucible server as a task repository.
- Build Mylyn queries to filter your reviews, based on the Crucible server filters or your own custom criteria.
- Manage the reviews in the Mylyn task list.
- Take advantage of Mylyn's context management to keep track of the files and other assets associated with your review.
Create a Crucible Review from within Eclipse

- Create a review for a committed changeset.
- Create a patch review for files not yet committed.
- Choose to start the review immediately or leave it in draft state.

Conduct your Code Reviews in your IDE
- See the review details, files and comments in an Eclipse view.
- Assign reviewers without leaving Eclipse.
- Create and reply to comments.
- Edit or remove comments.
- Open the source code file at the line that the comment refers to.
- Add changesets or a patch to an existing review.
- Summarise and close the review or perform other review workflow actions from within your IDE.

See the Review Comments in Your Code

- See the comments annotated in your file editor.
- Add comments directly in the editor.
Work on the Review in the Diff View Too

- Open a diff view to see the changes under review, with the comments highlighted.
- Add comments directly in the diff view too.
Receive Notification of Review Updates

- Receive notification of new reviews needing your attention.
- See what’s happening in your reviews, via the Atlassian Eclipse Connector’s integration with the standard Eclipse notification messages.

Bamboo Builds in Eclipse

Using the Atlassian Eclipse Connector, you can set up your Bamboo server as a Mylyn task repository. Now you can choose the builds you want to monitor and view the build results, build logs, changed files and failed tests inside your IDE.

Configure your Bamboo Server as a Mylyn Task Repository

- Configure your Bamboo server as a task repository.
- Choose the builds you want to monitor from within Eclipse.
Manage your Bamboo Builds in your IDE

- View the status of your selected builds.
- Right-click a build line to act upon it.
- Re-run the build.
- Add a label or comment to the build.
- Double-click the build to open it in an Eclipse view.
View the Build Logs

- Open the Bamboo build log in your Eclipse Console view.
**View the Changed Files**

- See the changes that triggered the build.

**Monitor the Failed Tests**

- See the test results for the selected build.
- Re-run the failed tests locally, to make sure your fixes work before breaking the build again.
- Create a new Mylyn task from the failure. This Mylyn task could be a JIRA issue.
**Receive Notification of Build changes**

- Receive build notifications via the Atlassian Eclipse Connector’s integration with the standard Eclipse notification messages.

**JIRA Issues in Eclipse**
The new connector incorporates the JIRA Mylyn plugin, which was previously available as a JIRA-only connector. Configure your JIRA server as a task repository. Use Mylyn's context management to keep track of the files and other assets associated with your issue.

**Use the Awesome JIRA Mylyn Plugin as Part of the New Connector**

- Configure your JIRA server as a task repository.
- Build Mylyn queries to filter your issues, based on your JIRA server filters or on custom criteria.
- Manage the issues via the Mylyn task list.
- Take advantage of Mylyn's context management to keep track of the files and other assets associated with your issue.
- Create a new JIRA issue and edit existing issues.
- Add and reply to comments.
- Receive notification of updated issues.

**Complete List of Fixes in this Release**

<table>
<thead>
<tr>
<th>JIRA Issues (54 issues)</th>
<th>Priority</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>JIRA Issues</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key</td>
<td>Summary</td>
<td></td>
</tr>
<tr>
<td>PLE-505</td>
<td>Release Engineering</td>
<td>📁</td>
</tr>
<tr>
<td>PLE-499</td>
<td>Pre 1.0 nits</td>
<td>📁</td>
</tr>
<tr>
<td>PLE-498</td>
<td>New review Wizard: minor nits</td>
<td>Closed</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>PLE-496</td>
<td>Recent change (ACC-31) seems to have disabled validating a connection with a Crucible server behind http authentication</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-495</td>
<td>Sort projects in Create Review wizard (project drop-down/combo)</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-489</td>
<td>Add support for Crucible servers being also FishEye servers</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-486</td>
<td>Installation fails with unsatisfied dependency error</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-484</td>
<td>Add Atlassian Preference (About) Page</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-482</td>
<td>Create Bamboo preference page</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-481</td>
<td>Please review Eclipse Connector 1.0 Configuration Guide</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-480</td>
<td>Fix spelling of &quot;Favorites&quot; on Bamboo repository settings screen</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-479</td>
<td>Please review Eclipse Connector 1.0 installation guide</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-478</td>
<td>Please review Eclipse Connector 1.0 release notes</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-477</td>
<td>Bamboo core/test not compiling due to commons changes</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-475</td>
<td>Smaller Icons for review editor</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-469</td>
<td>Fix compilation problems after recent refactoring made to ACC (Comment.getReplies)</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-466</td>
<td>Remove beta tags from UI</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-464</td>
<td>Nits discussed in the M3.5 meeting</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-462</td>
<td>Opening Crucible Editor blocks UI</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-460</td>
<td>Copying something to clipboard in the last screen of the review creation wizard overwrote my patch</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-456</td>
<td>Incorporate recent changes around ServerCfg -&gt; ServerData in Atlassian Connector Commons</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-454</td>
<td>Change presentation of &quot;reviewFiles&quot; that originate from a &quot;review from patch&quot;</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-453</td>
<td>Fix reflow/layout problems in CrucibleEditor</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-449</td>
<td>Change nightly builds to point to the Mylyn release update site</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-447</td>
<td>Referenced identifier 'com.atlassian.connector.eclipse.cruicible.ui.comment.annotation' in attribute 'annotationType' cannot be found</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-443</td>
<td>Add option to immediately start review while creating it from Eclipse</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-437</td>
<td>Test JIRA connector 3.1.1 release</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-422</td>
<td>File lookup of Subclipse connector does not work for our internal (subversive) repository</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-420</td>
<td>Hide inapplicable actions related to adding comments on a code review</td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-412</td>
<td>Provide Wizard for creating Review</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-410</td>
<td>opening Crucible editor leaves focus on Task List</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-400</td>
<td>Review editor does not get correctly updated when a user removes a reply to a comment</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-399</td>
<td>Create action for saving/submitting changes made in the ReviewEditor</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-398</td>
<td>Copy to clipboard a link to Fisheye from a file in Eclipse</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-397</td>
<td>Click through to Fisheye Browser from a file in Eclipse</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-395</td>
<td>The connector should support Subversive SVN Team provider</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-390</td>
<td>Refactor Popup Dialogs and Error Handling in DefaultTeamProvider / TeamUtilis</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-389</td>
<td>editor location jumps on refresh</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-388</td>
<td>Refactor Bamboo Actions</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-386</td>
<td>refactor AddGeneralCommentToFileAction</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-381</td>
<td>Clean up Exception handling in opening JUnit Action</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-380</td>
<td>user access rights are not respected while offering them add general file comment functionality</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-371</td>
<td>Add API nature to atlassian projects</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-368</td>
<td>Could not open JUnit Test Results</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-316</td>
<td>Post Drafts</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-315</td>
<td>Edit/Delete own comments (including drafts)</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-314</td>
<td>Edit Comments</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-299</td>
<td>Action: create Task from failed Build</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-250</td>
<td>create a basic rich editor for new reviews</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-245</td>
<td>provide a listing of all of the changesets to be added to the new review in the rich editor</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-239</td>
<td>provide a way of adding reviewers to the review in the new review rich editor</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-238</td>
<td>add an action to a changeset in the subclipse history view</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-173</td>
<td>Create Crucible review with patch (uncommitted code)</td>
<td>❌</td>
</tr>
<tr>
<td>PLE-172</td>
<td>Create Crucible review with committed files</td>
<td>❌</td>
</tr>
</tbody>
</table>
Atlassian Eclipse Connector 1.0 Beta Release Notes

23 March 2009

With pleasure, Atlassian presents the Atlassian Eclipse Connector 1.0 Beta.

Atlassian and Tasktop have collaborated to build a new connector with a radically changed look and feel and a wealth of new features. The resulting connector makes use of Mylyn’s task-focused interface to bring new level of integration between your Eclipse Mylyn IDE, your Crucible code reviews and your Bamboo builds. The new connector also includes the existing JIRA Mylyn plugin.

With the Atlassian Eclipse Connector, you can set up your JIRA, Bamboo and Crucible servers as Eclipse Mylyn task repositories. Monitor your Bamboo builds and re-run failed tests locally. Enjoy Crucible’s lightweight workflow to conduct code reviews within your IDE. The JIRA issues and Crucible reviews are seen as tasks, allowing you to make use of Mylyn’s powerful task and context management features. Instead of dropping out to view your email, you will now receive your issue, review and build notifications via the standard Eclipse notification popups.

Highlights of this Release:

- **Crucible Reviews in Eclipse**
  - Manage your Code Reviews via Mylyn’s Task-Focused Interface
  - Conduct your Code Reviews in your IDE
  - See the Review Comments in Your Code
  - Work on the Review in the Diff View Too
  - Receive Notification of Review Updates
- **Bamboo Builds in Eclipse**
  - Configure your Bamboo Server as a Mylyn Task Repository
  - Manage your Bamboo Builds in your IDE
  - View the Build Logs
  - View the Changed Files
  - Monitor the Failed Tests
  - Receive Notification of Build changes
- **JIRA Issues in Eclipse**
  - Use the Awesome JIRA Mylyn Plugin as Part of the New Connector

Don't have the Atlassian Eclipse Connector yet?

You can install the connector directly from the 'Software Updates' option in Eclipse. Follow our easy installation guide.

This is an open source project. The source code is available in our SVN repository.

⭐ We love your feedback! ⭐

Please log your issues, requests and votes. They help us decide what needs doing. It is easy to create an issue. (If you do not yet have an account, the system will prompt you to create one.)

<table>
<thead>
<tr>
<th>Upgrading from a previous version of the IDE Plugin for Eclipse</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you have previously installed an earlier version of the Atlassian IDE Plugin for Eclipse, please uninstall the earlier version. Then follow the installation guide to install the new Atlassian Eclipse Connector.</td>
</tr>
</tbody>
</table>

Crucible Reviews in Eclipse

Enjoy Crucible’s lightweight workflow to conduct code reviews within your IDE. Using the connector, you can handle your Crucible reviews as Eclipse Mylyn tasks and take advantage of Mylyn’s powerful context management features. In addition, you can perform much of the review workflow without leaving your IDE.

Manage your Code Reviews via Mylyn’s Task-Focused Interface

- Set up your Crucible server as a task repository.
- Build Mylyn queries to filter your reviews, based on the Crucible server filters or your own custom criteria.
- Manage the reviews in the Mylyn task list.
- Take advantage of Mylyn's context management to keep track of the files and other assets associated with your review.
Conduct your Code Reviews in your IDE

- See the review details, files and comments in an Eclipse view.
- Create and reply to comments.
- Summarise and close the review or perform other review workflow actions from within your IDE.
See the Review Comments in Your Code

- See the comments annotated in your file editor.
- Add comments directly in the editor.

Work on the Review in the Diff View Too

- Open a diff view to see the changes under review, with the comments highlighted.
- Add comments directly in the diff view too.
Receive Notification of Review Updates

- Receive notification of new reviews needing your attention.
- See what's happening in your reviews, via the Atlassian Eclipse Connector's integration with the standard Eclipse notification messages.

Bamboo Builds in Eclipse

Using the Atlassian Eclipse Connector, you can set up your Bamboo server as a Mylyn task repository. Now you can choose the builds you want to monitor and view the build results, build logs, changed files and failed tests inside your IDE.

Configure your Bamboo Server as a Mylyn Task Repository

- Configure your Bamboo server as a task repository.
- Choose the builds you want to monitor from within Eclipse.
Manage your Bamboo Builds in your IDE

- View the status of your selected builds.
- Right-click a build line to act upon it.
- Re-run the build.
- Add a label or comment to the build.
- Double-click the build to open it in an Eclipse view.
View the Build Logs

- Open the Bamboo build log in your Eclipse Console view.
**View the Changed Files**

- See the changes that triggered the build.

- **ECL-TST-375**
  - Project: Atlassian Eclipse Connector
  - Plant: Compile and Test
  - Build: 375

**Summary**

- Build ECL-TST-375 Failed.
- Build completed on Wed Mar 18 11:05:02 PDT 2009.
- Build took 18 minutes.
- Build reason: Manual build.

**Failed Tests**

- VersionedCommentDateComparatorTest: Compare
- CrucibleJUnitTest: Get Crucible Task from Id
- CrucibleJUnitTest: Get Crucible Task Repository From String

**Code Changes**

- [sminto] IN PROGRESS - issue PLE-277: add license to features
- [sminto] OPEN - issue PLE-218: Immediately show own comments in Rich Editor

**Build Log**

The build generated 143 lines (8 error lines). See the full build log for details.

**Failed Tests**

- BUILD FAILED

**Monitor the Failed Tests**

- See the test results for the selected build.
- Re-run the failed tests locally, to make sure your fixes work before breaking the build again.
- Create a new Mylyn task from the failure. This Mylyn task could be a JIRA issue.
Receive Notification of Build changes

- Receive build notifications via the Atlassian Eclipse Connector’s integration with the standard Eclipse notification messages.

JIRA Issues in Eclipse
The new connector incorporates the **JIRA Mylyn plugin**, which was previously available as a JIRA-only connector. Configure your **JIRA server** as a task repository. Use Mylyn's context management to keep track of the files and other assets associated with your issue.

**Use the Awesome JIRA Mylyn Plugin as Part of the New Connector**

- Configure your JIRA server as a task repository.
- Build Mylyn queries to filter your issues, based on your JIRA server filters or on custom criteria.
- Manage the issues via the Mylyn task list.
- Take advantage of Mylyn's context management to keep track of the files and other assets associated with your issue.
- Create a new JIRA issue and edit existing issues.
- Add and reply to comments.
- Receive notification of updated issues.

---

**Atlassian IDE Plugin 0.4 for Eclipse - Release Notes**

1 August 2008

With pleasure, Atlassian presents **version 0.4** of the Atlassian IDE Plugin for Eclipse.

This release brings further enhancements to the Bamboo support in the plugin. You can now connect to multiple Bamboo servers instead of just one. For each of your build plans, you can now show the full build log within the Eclipse IDE.

**Highlights of this Release**
There are now two Bamboo views: 'Bamboo Servers' and 'Bamboo Watched Plans'.

The 'Bamboo Watched Plans' view shows the information for your selected build plans, as in previous versions of the plugin.

The new 'Bamboo Servers' view allows you to add Bamboo servers and view the plans defined on each server.

You can now connect to multiple Bamboo servers instead of just one.
You can now show the full build log within the Eclipse IDE. Just select the build plan, then click the 'Show Build Log' icon in the toolbar.
Complete List of Fixes in the Atlassian IDE Plugin 0.4 for Eclipse

<table>
<thead>
<tr>
<th>JIRA Issues (4 issues)</th>
<th>Summary</th>
<th>Priority</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLE-51</td>
<td>Add new view to show defined bamboo servers</td>
<td><img src="https://www.atlassian.com/icons/status-up.png" alt=" " /></td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-49</td>
<td>Release notes for Eclipse plugin 0.4</td>
<td><img src="https://www.atlassian.com/icons/status-up.png" alt=" " /></td>
<td>Closed</td>
</tr>
<tr>
<td>PLE-45</td>
<td>Show full log for build</td>
<td><img src="https://www.atlassian.com/icons/status-up.png" alt=" " /></td>
<td>Resolved</td>
</tr>
<tr>
<td>PLE-40</td>
<td>Several icons in status bar</td>
<td><img src="https://www.atlassian.com/icons/status-up.png" alt=" " /></td>
<td>Resolved</td>
</tr>
</tbody>
</table>

Atlassian IDE Plugin 0.3 for Eclipse - Release Notes

Plugin version 0.3 requires Eclipse 3.4
Update on 25/7/2008: From version 0.3 of the Atlassian IDE Plugin, you will need Eclipse version 3.4.0 or later. For more information, see PLE-46.

18 July 2008

With pleasure, Atlassian presents version 0.3 of the Atlassian IDE Plugin for Eclipse. Here are the highlights of this release:

- A new Bamboo icon in the Eclipse status bar shows the summary status of builds: red means at least one build is currently failing; green means all builds are passing.
- The configuration screen allows you to select your favourite build plans as defined on the Bamboo server.
- You can now add labels and comments to your build plans (Bamboo 2 only).

Below is a list of all the fixes and improvements in this release.

The Atlassian IDE Plugin is available for IDEA too.

Take a look at the release notes for the IDEA version of the plugin.

Don't have the Atlassian IDE Plugin for Eclipse yet?
Follow our easy installation instructions.

This is an open source project. You can get the source code from our SVN repository.

The plugin's blog

From time to time, the plugin development team posts some information on the plugin's blog.

Complete List of Fixes in the Atlassian IDE Plugin 0.3 for Eclipse

<table>
<thead>
<tr>
<th>JIRA Issues (8 issues)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>PLE-39</td>
</tr>
<tr>
<td>PLE-38</td>
</tr>
<tr>
<td>PLE-34</td>
</tr>
<tr>
<td>PLE-32</td>
</tr>
<tr>
<td>PLE-23</td>
</tr>
<tr>
<td>PLE-22</td>
</tr>
<tr>
<td>PLE-16</td>
</tr>
<tr>
<td>PLE-9</td>
</tr>
</tbody>
</table>

Atlassian IDE Plugin 0.2 for Eclipse - Release Notes

3 July 2008

With pleasure, Atlassian presents version 0.2 of the Atlassian IDE Plugin for Eclipse.

Here are the highlights of this release:

- The configuration screen allows you to select your Bamboo build plans from a list, rather than typing them into a text box.
- The new 'Test Connection' button lets you verify the connection data you have entered, by connecting to the Bamboo server.
- The new 'Refresh' button lets you retrieve the list of build plans, using the most recent data entered into the form.
- A notification in the Eclipse status bar lets you know the status of background plugin activities, such as retrieving the list of plans or the build status.

Below is a list of all the fixes and improvements in this release.

The Atlassian IDE Plugin is available for IDEA too

Take a look at the release notes for the IDEA version of the plugin.

Don't have the Atlassian IDE Plugin for Eclipse yet?

Follow our easy installation instructions.

This is an open source project. You can get the source code from our SVN repository.

The plugin's blog

From time to time, the plugin development team posts some information on the plugin's blog.

Complete List of Fixes in the Atlassian IDE Plugin 0.2 for Eclipse

<table>
<thead>
<tr>
<th>JIRA Issues (5 issues)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
Atlassian IDE Plugin 0.2 for Eclipse - Upgrade Notes

Below are some essential notes on upgrading to version 0.2 of the Atlassian IDE Plugin for Eclipse. For details of the new features and improvements in this release, please read the Release Notes.

Upgrade Instructions

1. Please upgrade as usual. Instructions are available in the Upgrade Guide.
2. After upgrading, please change the name of your existing configuration file from `atlassian_eclipse_plugin.prefs` to `atlassian-eclipse-plugin.prefs` (replacing the underscores with hyphens).

Background Information

For the first time in this release, we built the release automatically using Maven with an Ant script. A side effect is that your plugin configuration data will be empty after upgrade. To retrieve the configuration data, change the file name as described above.

RELATED TOPICS

Atlassian IDE Plugin 0.2 for Eclipse - Release Notes

Atlassian IDE Plugin 0.1 for Eclipse - Release Notes

23 June 2008

With pleasure, Atlassian presents version 0.1 of the Atlassian IDE Plugin for Eclipse. Highlights of this release include:

- You can configure the polling interval that the plugin will use to monitor the Bamboo build plans.
- Bamboo will now notify you of build failures and successes via a popup. The popup's behaviour is configurable via the plugin’s Preferences window.
- You can change the order of the columns in the Bamboo view. The plugin will remember the column order the next time you start Eclipse.

The Atlassian IDE Plugin is available for IDEA too

Take a look at the release notes for the IDEA version of the plugin.

Don't have the Atlassian IDE Plugin for Eclipse yet?

Follow our easy installation instructions.

This is an open source project. You can get the source code from our SVN repository.

The plugin’s blog

The plugin’s team from time to time posts some info on the plugin’s blog

Complete List of Fixes in the Atlassian IDE Plugin 0.1 for Eclipse

<table>
<thead>
<tr>
<th>JIRA Issues (5 issues)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
</tr>
<tr>
<td>PLE-18</td>
</tr>
<tr>
<td>PLE-17</td>
</tr>
<tr>
<td>PLE-5</td>
</tr>
<tr>
<td>PLE-2</td>
</tr>
</tbody>
</table>
Configuring the Eclipse Connector

This page tells you how to set up your JIRA, Bamboo, Crucible and FishEye servers as task repositories in Eclipse.

On this page:

- Prerequisites
- Setting Up your Task Repositories
  - Setting Up a Bamboo Task Repository
  - Setting Up a Crucible Task Repository
  - Setting Up a FishEye Task Repository
  - Setting Up a JIRA Task Repository

Prerequisites

Before you can set up your task repositories for the Atlassian Connector for Eclipse, you will need to do the following:

- Install the Atlassian Connector for Eclipse as described in the installation guide.
- Check your version of Mylyn.

The Atlassian Connector for Eclipse makes use of Mylyn's task-focused interface. Mylyn is shipped with the more recent versions of Eclipse. The Atlassian Connector for Eclipse requires Mylyn 3.2.x or Mylyn 3.3.x.

- If you do not already have Mylyn, it will be installed automatically when you install one of the Atlassian Connector's features.
- If you already have Mylyn installed, please check your version of Mylyn. If your version of Mylyn is earlier than 3.2, please upgrade to Mylyn 3.3.x.
- The Atlassian Connector for Eclipse now includes the JIRA Mylyn connector, which was previously available as a JIRA-only connector.
- The Mylyn documentation includes a quick start guide.

Setting Up your Task Repositories

1. In Eclipse Mylyn, open the 'Task Repositories' view:
   - Click 'Window', 'Show View', 'Other'.
   - Open the 'Tasks' category and select the 'Task Repositories' view.
2. Click the 'Add Task Repository' icon.
3. The 'Add Task Repository' screen appears, as shown below:
4. Now you can select and configure one or more of the task repositories and configure them to match your server(s), as described below.
Once you have set up a repository, it will appear in the Eclipse 'Task Repositories' view, as shown in this screenshot:

---

**Setting Up a Bamboo Task Repository**

1. On the Eclipse 'Add Task Repository' screen, select the 'Bamboo' task repository type and click 'Next'.
2. The 'Bamboo Repository Settings' screen appears, as shown below:

   ![](Add_Task_Repository.png)

   - **Server** — The location (URL) of your Bamboo server.
   - **Label** — A descriptive name for your Bamboo server, e.g. 'Bamboo Extranet'.

3. Enter the following information:

   - **Server** — The location (URL) of your Bamboo server.
   - **Label** — A descriptive name for your Bamboo server, e.g. 'Bamboo Extranet'.
Connected — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.

User ID — The username you use to connect to your Bamboo server.

Password — Your password on the Bamboo server, matching the above username.

Save Password — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. Click the 'Refresh' button to verify the information you have entered. The connector will attempt to connect to the Bamboo server. It will also retrieve the latest list of build plans from the Bamboo server.

5. Select the plans that you want the connector to watch. You can either select the plans individually or you can click the 'Favourites' button to choose your favourite plans as defined on the Bamboo server.

6. Click 'Finish' to save the changes.

7. The new task repository appears in your Eclipse 'Task Repositories' view.

8. If necessary, you can now you can adjust the Bamboo settings.

   • Select 'Window', 'Preferences' then open the Bamboo settings panel as shown below:

   ![Bamboo settings panel](image)

   • Enter the 'Auto Refresh Interval Rate'. This is the number of minutes that the connector will wait between calls to fetch new information from your Bamboo server, when the connector is set to refresh automatically.

   • Tick or untick the 'Refresh Automatically' checkbox. Untick this option if you do not want the connector to poll the server automatically. In that case, you will need to refresh you Bamboo information manually by synchronising the Bamboo task repository.

9. Next step: To see your Bamboo builds, open the 'Bamboo' view. (Select 'Window', 'Show View', 'Other', then open the 'Atlassian' folder.)

   You can configure one or more Bamboo repositories, i.e. you can connect to more than one Bamboo server.

Bamboo builds are shown in a Bamboo-specific view

Bamboo has its own 'Bamboo' view. This is different from JIRA and Crucible, where issues and reviews are included into your 'Task List' view. Next, take a look at the Bamboo view.

Setting Up a Crucible Task Repository

- Single server for both Crucible and FishEye = single task repository

   If you have a single server for both FishEye and Crucible, you should set up the Crucible task repository and then select the 'Crucible Server Contains FishEye Instance' option as described below. Due to Mylyn requirements, you cannot set up two separate task repositories for the same URL.

1. On the Eclipse 'Add Task Repository' screen, select the 'Crucible' task repository type and click 'Next'.

2. The 'Crucible Repository Settings' screen appears, as shown below:
3. Enter the following information:
   - **Server** — The location (URL) of your Crucible server.
   - **Label** — A descriptive name for your Crucible server, e.g. ‘Extranet Crucible’.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your Crucible server.
   - **Password** — Your password on the Crucible server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

   If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. Click the ‘Validate Settings’ button to verify the information you have entered.

5. Choose a ‘Review Activation’ setting. These settings determine what happens when you open a source code file from within the Crucible review editor in Eclipse. For example, you may open a Crucible review within Eclipse. The review will open in an Eclipse editor view, and its ‘Review Files’ section will show the files in the review. You can then click a file to open it in an Eclipse editor. The options are as follows:
   - **Always** — The Crucible review will be automatically activated if you open a file from the Crucible review editor.
   - **Never** — The Crucible review will never be automatically activated.
   - **Prompt** (default) — When you open a file from the Crucible review editor, the connector will ask you whether you want to activate the Crucible review. This is the default setting.

6. If your Crucible server is linked to a FishEye server:
   - Put a tick in the checkbox labelled ‘Crucible Server Contains FishEye Instance’.
   - Set up your FishEye mappings, as described in the FishEye section of this documentation.

7. Click ‘Finish’ to save the changes.

8. Eclipse Mylyn will prompt you to add a new query for the new Crucible repository. This is where you will choose your Crucible filter, to determine which reviews appear in your task list. You can do this now or skip this step and do it later. You can also add more queries later.


   You can configure one or more Crucible repositories, i.e. you can connect to more than one Crucible server.

---

**Setting Up a FishEye Task Repository**
If you have a single server for both FishEye and Crucible, you should set up the Crucible task repository and then select the 'Crucible Server Contains FishEye Instance' option as described above. Due to Mylyn requirements, you cannot set up two separate task repositories for the same URL.

If you do not have a Crucible server, you can hook up your FishEye server as an Eclipse task repository without the additional Crucible functionality.

1. On the Eclipse 'Add Task Repository' screen, select the 'FishEye' task repository type and click 'Next'.
2. The 'FishEye Repository Settings' screen appears, as shown below:

![FishEye Repository Settings](image.png)

3. Enter the following information:
   - **Server** — The location (URL) of your FishEye server.
   - **Label** — A descriptive name for your FishEye server, e.g. 'FishEye Atlassian Developer'.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your FishEye server.
   - **Password** — Your password on the FishEye server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

   If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. Click the 'Validate Settings' button to verify the information you have entered.
5. Click 'Finish' to save the changes.
6. The new task repository appears in your Eclipse 'Task Repositories' view.
7. Now you need to map your source code to your FishEye repository and FishEye project.
   - Select 'Window', 'Preferences' then open the 'Repository Mappings' panel as shown below:
Click the ‘Add’ button to add a new mapping. The ‘Add Mapping’ screen appears, as shown below:

- Enter the ‘SCM Path’ — Supply the path to your source code as used locally by the project.
- Select the FishEye/Crucible ‘Server’ as configured in your task repositories. (See instructions above.)
- The connector will retrieve a list of FishEye repositories from the server you have supplied. Select the appropriate ‘Source Repository’ to map your local project to the FishEye repository.

Some notes about the FishEye integration:

- You can configure one or more FishEye repositories, i.e. you can connect to more than one FishEye server.
- We have chosen to configure FishEye servers as standard Mylyn task repositories. They are not really regular repositories like JIRA or Crucible, because there are no tasks associated with a FishEye task repository. We are using the task repository concept to take advantage of various infrastructural facilities available in Mylyn and to make user experience more consistent with the rest of the Atlassian products. See More about Configuring FishEye Repositories in Eclipse.

Setting Up a JIRA Task Repository

1. On the Eclipse ‘Add Task Repository’ screen, select the ‘JIRA’ task repository type and click ‘Next’.
2. The ‘JIRA Repository Settings’ screen appears, as shown below:
3. Enter the following information:
   - **Server** — The location (URL) of your JIRA server.
   - **Label** — A descriptive name for your JIRA server, e.g., ‘JAC’.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your JIRA server.
   - **Password** — Your password on the JIRA server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.
   - **If** you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. If necessary, you can change the additional settings. If in doubt, leave these settings as set by default.
   - The 'Task Editor Settings' section is used to configure the markup language used by the task editor when Mylyn’s WikiText extension is installed. By default, the Atlassian Connector for Eclipse will set the markup language to ‘Confluence’, so that
WikiText recognises Confluence wiki markup as used by JIRA. See the Mylyn WikiText User Guide for more information.

5. Click 'Finish' to save the changes.

6. Eclipse Mylyn will prompt you to add a new query for the new JIRA repository. This is where you will choose your JIRA filter, to determine which issues appear in your task list. You can do this now or skip this step and do it later. You can also add more queries later.

7. The new task repository appears in your Eclipse 'Task Repositories' view. Your JIRA issues will appear in your 'Task List' view.

You can configure one or more JIRA repositories, i.e. you can connect to more than one JIRA server.

RELATED TOPICS

Installing the Eclipse Connector
IDE Connector Documentation

Collecting Usage Statistics for the Eclipse Connector

By default, the Atlassian Connector for Eclipse will collect information on the usage of the connector and send the information to Atlassian. We collect and analyse this information to better understand the usage of the connector, so that we can continue to develop a better product that meets your needs. You can choose to enable or disable the collection of information at any time.

On this page:
- Enabling or Disabling the Collection of Information
- Submitting the Usage Data Manually
- Information Collected
- Mechanism Used to Send Information
- Another Way to Send Feedback

Enabling or Disabling the Collection of Information

We understand, of course, that some people do not wish to have any information collected from them. You can enable or disable the collection of information at any time, and set other preferences.

To set your usage data preferences,

1. In Eclipse, open the 'Window' menu and select 'Preferences'.
2. Open the 'Atlassian Connector' folder.
3. Select 'Usage Data'.

Screenshot 1: Configuration preferences for usage data collection

Submitting the Usage Data Manually
You can choose to submit your usage data manually, rather than allowing automatic submission.

To submit usage data manually,

1. In Eclipse, open the ‘File’ menu and select ‘New’, ‘Other’.
2. The ‘Select a Wizard’ screen appears, as shown in screenshot 2 below. Open the ‘Other’ folder.
3. Select ‘Usage Data Submission for Atlassian Connector for Eclipse’ and click ‘Next’.
4. The ‘Usage Data Submission’ screen appears, as shown in screenshot 3 below. Click the ‘Finish’ button to send the data to Atlassian.

Screenshot 2: Starting the Usage Data Submission wizard

![Screenshot 2](image)

Screenshot 3: Submitting the usage data manually

![Screenshot 3](image)
Information Collected

We track general events in Eclipse, such as when you open a task, open a Bamboo build, open a Crucible review, get the Bamboo build details, get a FishEye resource link, get Bamboo build output, and so on.

We store things like:

- Event start and end dates.
- Event originator, such as the workbench window, command, toolbar, etc.
- Event name — usually the class name of the handler.
- Task type (jira, crucible, etc), for tasks opened by the user.

The information we collect is anonymous and cannot be used to identify you. Private information, such as task key and title, is obfuscated by default. If you choose not to obfuscate data, the information collected will include things like task key, task title, and in some cases a partial path name.

Mechanism Used to Send Information

The connector collects the information in a monitor-log.xml file, located in your Eclipse workspace directory as specified in your configuration settings (see above). The connector sends the data to Atlassian as a zipped XML file, periodically at the time interval specified in your configuration settings (see above).

Another Way to Send Feedback

Even if you decide not to send usage data through to us, we would still appreciate any feedback, comments, or suggestions you may have via our JIRA issue tracker or our forums.

RELATED TOPICS

Configuring the Eclipse Connector

Using JIRA in the Eclipse Connector

The Atlassian Connector for Eclipse allows you to monitor, add and update JIRA issues while remaining in your integrated development environment.

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the Installation Guide.

- Configuring your JIRA Options in Eclipse
- Working with JIRA Issues in Eclipse

RELATED TOPICS
Configuring your JIRA Options in Eclipse

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the Installation Guide.

On this page:

- Adding a Task Repository
- Setting Up a JIRA Task Repository

Adding a Task Repository

1. In Eclipse Mylyn, open the 'Task Repositories' view:
   - Click 'Window', 'Show View', 'Other'
   - Open the 'Tasks' category and select the 'Task Repositories' view.
2. Click the 'Add Task Repository' icon.
3. The 'Add Task Repository' screen appears, as shown below:

   ![Add Task Repository Screen]

4. Now you can select and configure one or more of the task repositories and configure them to match your server(s), as described below.

   Once you have set up a repository, it will appear in the Eclipse 'Task Repositories' view, as shown in this screenshot:

   ![Task Repositories View]

Setting Up a JIRA Task Repository

1. On the Eclipse 'Add Task Repository' screen, select the 'JIRA' task repository type and click 'Next'.
2. The 'JIRA Repository Settings' screen appears, as shown below:
3. Enter the following information:
   - **Server** — The location (URL) of your JIRA server.
   - **Label** — A descriptive name for your JIRA server, e.g., 'JAC'.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your JIRA server.
   - **Password** — Your password on the JIRA server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

4. If necessary, you can change the additional settings. If in doubt, leave these settings as set by default.
   - The 'Task Editor Settings' section is used to configure the markup language used by the task editor when Mylyn's WikiText extension is installed. By default, the Atlassian Connector for Eclipse will set the markup language to 'Confluence', so that
WikiText recognises Confluence wiki markup as used by JIRA. See the Mylyn WikiText User Guide for more information.

5. Click ‘Finish’ to save the changes.

6. Eclipse Mylyn will prompt you to add a new query for the new JIRA repository. This is where you will choose your JIRA filter, to determine which issues appear in your task list. You can do this now or skip this step and do it later. You can also add more queries later.


You can configure one or more JIRA repositories, i.e. you can connect to more than one JIRA server.

RELATED TOPICS

Working with JIRA Issues in Eclipse
Installation and Upgrade Guide for the Eclipse Connector

Working with JIRA Issues in Eclipse

The Atlassian Connector for Eclipse allows you to monitor, add and update JIRA issues while remaining in your integrated development environment.

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the installation guide, and defined a task repository for at least one JIRA server, as described in the configuration guide.

On this page:

- User Guide
- Summary of What You Can Do
- Creating a query
- Creating a new issue

User Guide

For information on setting up your JIRA server in Eclipse, please read the configuration guide.

Summary of What You Can Do

Below are the highlights of viewing and acting upon JIRA issues within Eclipse. Follow the links to the relevant sections of the user guide.

Creating a query

The first step is to connect to a JIRA repository.

To connect to a JIRA repository,

1. Make the Task Repositories view visible using the menu Window > Show View > Other and select Tasks -> Task Repositories.
2. Click the ‘Add Task Repository’ icon in the toolbar of the view.

A wizard for setting the repository properties is displayed.
3. Choose JIRA (supports 3.4 and later) from the list of connectors and select Next. The ‘JIRA Repository Settings’ screen will display.
4. Enter the URL of your JIRA instance as the ‘Server’ URL (e.g. ‘http://jira.atlassian.com’).
5. Enter the label for the JIRA instance (e.g. ‘JAC’). This will display in your Eclipse IDE for the JIRA instance.
7. Click the ‘Validate Settings’ button to verify that the entered credentials are valid.
8. Select Finish to create the repository. You will then be prompted to create a query.

To create a new query,
1. ‘Create query using form’ will be selected by default. Click Next. On the query creation, Mylyn will download the repository configuration which can take a few minutes. The ‘Edit Repository Query’ window will display. Enter the following information:
1. Enter a **Query Title** that for your query.
2. Choose search parameters that define the query.
3. Click the **Finish** button to add the query to your task list.

The Java perspective displays the task list in the upper right section of the workbench. If it is not visible go to **Window > Show View > Other** and select **Tasks > Task List**. The task list shows the list of tasks that match the query. The blue arrow icons indicate that the task have incoming changes. Double clicking a task will bring up the task editor.

*Screenshot: Task List*
The task editor displays the attributes, description and comments of a JIRA issue and allows you to edit the issue.

Screenshot: Task Editor

Creating a new issue

To create a new issue,

1. Click the 'New Task' icon in toolbar of the Task List. The 'New Task' wizard will display.
2. Choose your repository (e.g. 'JAC') from the list of repositories.
3. Select Next to and choose a project from the list of projects.
4. Select Finish to open the editor for entering task details.

**RELATED TOPICS**

- Installation and Upgrade Guide for the Eclipse Connector
- Configuring the Eclipse Connector
- Using JIRA in the Eclipse Connector
- Using Bamboo in the Eclipse Connector
- Using FishEye in the Eclipse Connector
- Using Crucible in the Eclipse Connector
- Eclipse Connector FAQ
- Eclipse Connector Resources

Refer to the JIRA documentation for more information about JIRA functionality.

**Using Bamboo in the Eclipse Connector**

The Atlassian Connector for Eclipse gives you Bamboo build information right there in your integrated development environment.

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the Installation Guide.

- Configuring your Bamboo Options in Eclipse
- Working with Bamboo Builds in Eclipse
  - Monitoring Build Statuses in Eclipse
  - Viewing Build Details in Eclipse
  - Running a Build from within Eclipse
  - Viewing Test Results in Eclipse
  - Viewing a Build Log in Eclipse
  - Adding a Task from a Failed Build in Eclipse

**RELATED TOPICS**

Installation and Upgrade Guide for the Eclipse Connector

Refer to the Bamboo documentation for more information about Bamboo builds.

**Configuring your Bamboo Options in Eclipse**

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the Installation Guide.

On this page:

- Adding a Task Repository
- Setting Up a Bamboo Task Repository

**Adding a Task Repository**

1. In Eclipse Mylyn, open the ‘Task Repositories’ view:
   - Click ‘Window’, ‘Show View’, ‘Other’.
   - Open the ‘Tasks’ category and select the ‘Task Repositories’ view.
2. Click the ‘Add Task Repository’ icon.
3. The ‘Add Task Repository’ screen appears, as shown below:
4. Now you can select and configure one or more of the task repositories and configure them to match your server(s), as described below.

Once you have set up a repository, it will appear in the Eclipse 'Task Repositories' view, as shown in this screenshot:

```
<Console> Bamboo | Task Repositories | Task List

Local
Bamboo developer
Eclipse.org
Atlassian
STAC Crucible
FishEye
JIRA
```

Setting Up a Bamboo Task Repository

1. On the Eclipse 'Add Task Repository' screen, select the 'Bamboo' task repository type and click 'Next'.
2. The 'Bamboo Repository Settings' screen appears, as shown below:
3. Enter the following information:
   * **Server** — The location (URL) of your Bamboo server.
   * **Label** — A descriptive name for your Bamboo server, e.g. ‘Bamboo Extranet’.
   * **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   * **User ID** — The username you use to connect to your Bamboo server.
   * **Password** — Your password on the Bamboo server, matching the above username.
   * **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

   If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. Click the ‘Refresh’ button to verify the information you have entered. The connector will attempt to connect to the Bamboo server. It will also retrieve the latest list of build plans from the Bamboo server.

5. Select the plans that you want the connector to watch. You can either select the plans individually or you can click the ‘Favourites’ button to choose your favourite plans as defined on the Bamboo server.

6. Click ‘Finish’ to save the changes.

7. The new task repository appears in your Eclipse ‘Task Repositories’ view.

8. If necessary, you can now you can adjust the Bamboo settings.
   * Select ‘Window’, ‘Preferences’ then open the Bamboo settings panel as shown below:
Enter the 'Auto Refresh Interval Rate'. This is the number of minutes that the connector will wait between calls to fetch new information from your Bamboo server, when the connector is set to refresh automatically.

Tick or untick the 'Refresh Automatically' checkbox. Untick this option if you do not want the connector to poll the server automatically. In that case, you will need to refresh your Bamboo information manually by synchronising the Bamboo task repository.

9. Next step: To see your Bamboo builds, open the 'Bamboo' view. (Select 'Window', 'Show View', 'Other', then open the 'Atlassian' folder.)

You can configure one or more Bamboo repositories, i.e. you can connect to more than one Bamboo server.

Bamboo builds are shown in a Bamboo-specific view
Bamboo has its own 'Bamboo' view. This is different from JIRA and Crucible, where issues and reviews are included into your 'Task List' view. Next, take a look at the Bamboo view.

RELATED TOPICS

Working with Bamboo Builds in Eclipse
Installation and Upgrade Guide for the Eclipse Connector

Working with Bamboo Builds in Eclipse

With the Atlassian Connector for Eclipse, you can monitor and act upon your Bamboo builds from within your Eclipse IDE.

Prerequisites

Please make sure that you have installed the Atlassian Connector for Eclipse, as described in the Installation Guide, and defined at least one Bamboo server, as described in the Configuration Guide.

Summary of What You Can Do

Below are the highlights of viewing and acting upon Bamboo builds within Eclipse. Follow the links to the relevant sections of the user guide.

- View a list of the builds you are monitoring, in the Bamboo view in Eclipse.
- Receive notification of failed builds and other build changes.
- Open the Bamboo build details in an Eclipse editor.
- Open the Bamboo build details in your web browser, displaying the Bamboo web interface.
- Run a build on the Bamboo server.
- View a Bamboo build log.
- View test results.
- View changed files in the build.
- Comment on a Bamboo build.
- Label a Bamboo build.
- Add a new task based on a failed build.

Opening the Bamboo View in Eclipse

Bamboo has its own 'Bamboo' view in Eclipse. This is different from JIRA and Crucible, where issues and reviews are included into your 'Task List' view.

To open the 'Bamboo' view,
1. In the Eclipse toolbar, select 'Window', 'Show View', 'Other'. See screenshot 1.
2. The 'Show View' popup window appears, as shown in screenshot 2. Open the 'Atlassian' folder and click 'Bamboo'.
3. The Bamboo view appears, as shown in screenshot 3.

Screenshot 1: The Eclipse 'Show View' menu

![Screenshot 1: The Eclipse 'Show View' menu](image1)

Screenshot 2: The Eclipse 'Show View' popup

![Screenshot 2: The Eclipse 'Show View' popup](image2)

Screenshot 3: The Bamboo view

![Screenshot 3: The Bamboo view](image3)
Next, take a look at the Bamboo build details, as displayed in an Eclipse editor.

**RELATED TOPICS**

- Installation and Upgrade Guide for the Eclipse Connector
- Configuring the Eclipse Connector
- Using JIRA in the Eclipse Connector
- Using Bamboo in the Eclipse Connector
- Using FishEye in the Eclipse Connector
- Using Crucible in the Eclipse Connector
- Eclipse Connector FAQ
- Eclipse Connector Resources

Refer to the Bamboo documentation for more information about Bamboo builds.

**Monitoring Build Statuses in Eclipse**

Once you have opened the Bamboo view in Eclipse, the Bamboo icon at the top of the view shows the summary status of your builds. If at least one build is currently failing, the icon has a red cross.

*Screenshot: Bamboo status icon at the top of the Bamboo view*
In addition, when the build status changes (e.g. a build fails or some other change occurs), an Eclipse notification popup window appears.

**Screenshot: Bamboo build notification**

![Screenshot: Bamboo build notification](image)

Next, take a look at the Bamboo build details, as displayed in an Eclipse editor.

**RELATED TOPICS**

- Working with Bamboo Builds in Eclipse
  - Monitoring Build Statuses in Eclipse
  - Viewing Build Details in Eclipse
  - Running a Build from within Eclipse
  - Viewing Test Results in Eclipse
  - Viewing a Build Log in Eclipse
  - Adding a Task from a Failed Build in Eclipse

**Viewing Build Details in Eclipse**

Once you have opened the Bamboo view in Eclipse, you can see the details of a specific build.

To view the details of a Bamboo build,
1. Open the Bamboo view:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears. Open the 'Atlassian' folder and click 'Bamboo'. (If you have trouble finding the options, try the full instructions with screenshots.)
2. Double-click the line showing the build you want, or right-click the line and select 'Open'.
3. The build details appear in an Eclipse editor, as shown in the screenshot below.

Screenshot: Viewing Bamboo build details

The Bamboo build details screen opens in an Eclipse editor. It displays the following:

- The build summary.
- The number of tests in the build, with details of the failed tests. See Viewing Test Results in Eclipse.
- The code changes included in the build.
- A summarised build log, with the option to see the full log. See Viewing a Build Log in Eclipse.
- Further options to:
  - Run the build on the Bamboo server. See Running a Build from within Eclipse.
  - Label the build. See the diagram above.
  - Add a comment to the build. See the diagram above.
  - Add a new task based on the failed build. See Adding a Task from a Failed Build in Eclipse.
  - Open the build in your browser, displaying Bamboo’s web interface. See the diagram above.

Next, take a look at viewing test results.

RELATED TOPICS

Working with Bamboo Builds in Eclipse

- Monitoring Build Statuses in Eclipse
- Viewing Build Details in Eclipse
- Running a Build from within Eclipse
- Viewing Test Results in Eclipse
- Viewing a Build Log in Eclipse
• Adding a Task from a Failed Build in Eclipse

Running a Build from within Eclipse

Once you have opened the Bamboo view in Eclipse, you can re-run a specific build. This will kick off the build on the Bamboo server.

To run a Bamboo build,

1. Open the Bamboo view:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears. Open the 'Atlassian' folder and click 'Bamboo'. (If you have trouble finding the options, try the full instructions with screenshots.)
2. Choose one of the following ways to run the build:
   - From the Bamboo view:
     a. Right-click the line showing the build you want, and select 'Run Build'.
   - Or, also from the Bamboo view:
     a. Click the 'Run Build on Server' icon in the view's toolbar.
     b. Or open the Bamboo build details screen:
        a. Double-click or right-click the line showing the build you want, on the Bamboo view, to open the Bamboo build details.
        b. The Bamboo build details are displayed in an Eclipse editor. Click the 'Run Build on Server' icon in the toolbar.

Screenshot: Various ways to run a Bamboo build

Next, take a look at viewing test results.

RELATED TOPICS
- Working with Bamboo Builds in Eclipse
  - Monitoring Build Statuses in Eclipse
Viewing Test Results in Eclipse

Once you have opened the Bamboo view in Eclipse, you can view the results of the tests in a specific build.

To view test results,

1. Open the Bamboo view:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears. Open the 'Atlassian' folder and click 'Bamboo'. (If you have trouble finding the options, try the full instructions with screenshots.)

2. Choose one of the following ways to view the test results — all methods are illustrated in diagram 1 below:
   - From the Bamboo view:
     a. Right-click the line showing the build you want, and select 'Show Test Results'.
   - Or, also from the Bamboo view:
     a. Select the line showing the build you want.
     b. Click the JUnit icon $\text{JUnit}$ in the view's toolbar.
   - Or open the Bamboo build details screen:
     a. Double-click or right-click the line showing the build you want, on the Bamboo view, to open the Bamboo build details.
     b. The Bamboo build details are displayed in an Eclipse editor. Click the JUnit icon $\text{JUnit}$ in the toolbar.

3. The JUnit view opens, as shown in diagram 2 below. Now you can use the JUnit functionality to:
   - Select a file and open it.
   - Re-run a test.
   - Debug the code.
   - Copy the failure trace information to your clipboard.

Diagram 1: Various ways to view the results of tests in a Bamboo build
Diagram 2: Test results displayed in JUnit

Next, take a look at viewing a build log.

RELATED TOPICS

Working with Bamboo Builds in Eclipse

- Monitoring Build Statuses in Eclipse
- Viewing Build Details in Eclipse
- Running a Build from within Eclipse
- Viewing Test Results in Eclipse
- Viewing a Build Log in Eclipse
- Adding a Task from a Failed Build in Eclipse

Viewing a Build Log in Eclipse

Once you have opened the Bamboo view in Eclipse, you can view the full log for a specific build.

To view a full build log,

1. Open the Bamboo view:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears. Open the 'Atlassian' folder and click 'Bamboo'. (If you have trouble finding the options, try the full instructions with screenshots.)
2. Choose one of the following ways to view the build log — all methods are illustrated in diagram 1 below:
   - From the Bamboo view:
     a. Right-click the line showing the build you want, and select 'Show Build Log'.
     b. Click the Show Build Log icon in the view's toolbar.
   - Or open the Bamboo build details screen:
     a. Double-click or right-click the line showing the build you want, on the Bamboo view, to open the Bamboo build details.
     b. The Bamboo build details are displayed in an Eclipse editor. Click the Show Build Log icon or the text 'full build log'.
3. The build log opens in the Eclipse console, as shown in screenshot 2 below.
Diagram 1: Various ways to open a build log

Screenshot 2: Build log displayed in Eclipse console
Next, take a look at adding a task based on a failed build.

RELATED TOPICS

Working with Bamboo Builds in Eclipse

- Monitoring Build Statuses in Eclipse
- Viewing Build Details in Eclipse
- Running a Build from within Eclipse
- Viewing Test Results in Eclipse
- Viewing a Build Log in Eclipse
- Adding a Task from a Failed Build in Eclipse

Adding a Task from a Failed Build in Eclipse

Once you have opened the Bamboo view in Eclipse, you can create a new Mylyn task based on the failure of a specific build. This task could be an JIRA issue, for example.

To add a task from a failed build,
1. Open the Bamboo view:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears. Open the 'Atlassian' folder and click 'Bamboo'. (If you have trouble finding the options, try the full instructions with screenshots.)
2. Choose one of the following ways to add a task — all methods are illustrated in diagram 1 below:
   - From the Bamboo view:
     a. Right-click the line showing the build you want, and select 'New Task From Failed Build'.
     b. The Bamboo build details are displayed in an Eclipse editor. Click the New Task From Failed Build icon in the toolbar.
3. The 'New Task' dialogue opens. Choose a task repository, to determine the task type. This task can be a JIRA issue, for example, as illustrated below. We have chosen a JIRA repository, as shown in screenshot 2.
4. Click 'Next'.
5. The 'New JIRA Task' dialogue opens, as shown in screenshot 3 below. Select the project for your task/issue.
6. Click 'Finish'.
7. The JIRA issue opens in the Eclipse editor, as shown in screenshot 4 below. Many of the issue details are already present, based on the details of the failed build. Enter or update any further details as required.
8. Click 'Submit'.

Diagram 1: Various ways to add a task from a failed Bamboo build

Screenshot 2: Selecting a task repository for the new task
Size parameter must be a number (optionally followed by 'px', 'pt' or 'em').

Screenshot 3: Selecting a project for the new task
Size parameter must be a number (optionally followed by 'px', 'pt' or 'em').
Using FishEye in the Eclipse Connector

Using the Atlassian Connector for Eclipse, you can move quickly between your IDE and your FishEye view of your source repository.

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the Installation Guide.

- Configuring your FishEye Options in Eclipse
- More about Configuring FishEye Repositories in Eclipse
- Working with your FishEye Repository View in Eclipse

RELATED TOPICS

Installation and Upgrade Guide for the Eclipse Connector

Refer to the FishEye documentation for more information about FishEye repository insight.

Configuring your FishEye Options in Eclipse

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the Installation Guide.

On this page:

- Adding a Task Repository
- Setting Up a FishEye Task Repository

Adding a Task Repository

1. In Eclipse Mylyn, open the 'Task Repositories' view:
   - Click 'Window', 'Show View', 'Other'
   - Open the 'Tasks' category and select the 'Task Repositories' view.
2. Click the 'Add Task Repository' icon.
3. The 'Add Task Repository' screen appears, as shown below:
3. Now you can select and configure one or more of the task repositories and configure them to match your server(s), as described below.

Setting Up a FishEye Task Repository

4. Once you have set up a repository, it will appear in the Eclipse 'Task Repositories' view, as shown in this screenshot:

5. Setting Up a FishEye Task Repository

   - Single server for both Crucible and FishEye = single task repository
     - If you have a single server for both FishEye and Crucible, you should set up the Crucible task repository and then select the 'Crucible Server Contains FishEye Instance' option as described above. Due to Mylyn requirements, you cannot set up two separate task repositories for the same URL.

   - If you do not have a Crucible server, you can hook up your FishEye server as an Eclipse task repository without the additional Crucible functionality.

   1. On the Eclipse 'Add Task Repository' screen, select the 'FishEye' task repository type and click 'Next'.
   2. The 'FishEye Repository Settings' screen appears, as shown below:
3. Enter the following information:
   - **Server** — The location (URL) of your FishEye server.
   - **Label** — A descriptive name for your FishEye server, e.g. ‘FishEye Atlassian Developer’.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your FishEye server.
   - **Password** — Your password on the FishEye server, matching the above username.
   - **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

   If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. Click the ‘Validate Settings’ button to verify the information you have entered.
5. Click ‘Finish’ to save the changes.
6. The new task repository appears in your Eclipse ‘Task Repositories’ view.
7. Now you need to map your source code to your FishEye repository and FishEye project.
   - Select ‘Window’, ‘Preferences’ then open the ‘Repository Mappings’ panel as shown below:
Click the 'Add' button to add a new mapping. The 'Add Mapping' screen appears, as shown below:

- Enter the 'SCM Path' — Supply the path to your source code as used locally by the project.
- Select the FishEye/Crucible 'Server' as configured in your task repositories. (See instructions above.)
- The connector will retrieve a list of FishEye repositories from the server you have supplied. Select the appropriate 'Source Repository' to map your local project to the FishEye repository.

Some notes about the FishEye integration:

- You can configure one or more FishEye repositories, i.e. you can connect to more than one FishEye server.
- We have chosen to configure FishEye servers as standard Mylyn task repositories. They are not really regular repositories like JIRA or Crucible, because there are no tasks associated with a FishEye task repository. We are using the task repository concept to take advantage of various infrastructural facilities available in Mylyn and to make user experience more consistent with the rest of the Atlassian products. See More about Configuring FishEye Repositories in Eclipse.

RELATED TOPICS

Installation and Upgrade Guide for the Eclipse Connector

More about Configuring FishEye Repositories in Eclipse
This page contains additional information that you may find useful when setting up your FishEye server for the Atlassian Connector for Eclipse. You will find the step-by-step instructions in Configuring the Eclipse Connector.

On this page:
- Background to the FishEye Configuration and Mappings
- Limitations

Background to the FishEye Configuration and Mappings

As an Eclipse plugin, the Atlassian connector needs additional configuration for the following reasons:

- The connector needs to know which FishEye server is serving the selected source file, and in which FishEye repository the file exists.
- The connector needs to know the current revision (or version) of the local file, in order to take the user to the proper place in the history on the FishEye server.
- Moreover files may come from various places in the repository, such as trunk, branches, tags, modules for CVS, etc.
- FishEye and Eclipse may use different access protocols to access the same source. For example, FishEye may use `file://` for SVN, while Eclipse uses `https://`.
- As a developer, you may have many projects open in a single Eclipse workspace. These projects come from different source repositories (SVN,CVS, etc) that are potentially indexed by different FishEye servers.

It would be time-consuming and unnecessary to configure FishEye for each project, because often many projects come from the same repository and are served by single FishEye instance and maybe by one or two FishEye repositories in that instance.

The solution:

- Configure the FishEye mappings not per project but per SCM setup (path) used locally by the project.
- Combine the selected SCM path (as seen locally in Eclipse) with the FishEye server (as configured in the Mylyn task repositories) and the FishEye repository from the selected FishEye server.
- Given a source file, the Atlassian Connector for Eclipse checks the SCM path from which the file was checked out, and finds the appropriate mapping (the combination as defined above). Then it knows which FishEye server and which FishEye repository the selected file belongs to.

Limitations

Please take note of the following:

- The configuration does not keep track of re-configuration in the Mylyn task repositories. If the mapped FishEye server is removed from the list of task repositories, your FishEye functionality may stop working.
- The file may be deleted on the server side or FishEye may be out of sync with the SCM repository, so you may get an HTTP 404 error.
- Please refer to the notes on support for specific source control repositories.

RELATED TOPICS
Configuring the Eclipse Connector
IDE Connector Documentation

Working with your FishEye Repository View in Eclipse

Using the Atlassian Connector for Eclipse, you can move quickly between your IDE and your FishEye view of your source repository.

Prerequisites

Please make sure that you have installed the Atlassian Connector for Eclipse, as described in the installation guide, and defined at least one FishEye or Crucible+FishEye server, as described in the configuration guide.

Make sure that you have mapped your source code to your FishEye repository and FishEye project, as described in the configuration guide. The options described below will not work if the FishEye mappings are not configured correctly. You will receive an HTTP 404 error ('404 – Not Found') when you attempt to open the FishEye resource.

Opening a File in FishEye from Eclipse

To open a file from Eclipse in FishEye, or to send a FishEye link to your colleagues,

- Right-click the file in Eclipse and select one of the following:
  - 'Team' then 'Open in FishEye'.
  - Or 'Team' then 'Copy FishEye Link to Clipboard'.

This works from a number of Eclipse views. The only requirement is that the object you right-click represents a resource under version control with the FishEye mapping correctly configured. Here are some examples of where it will work:
Below we illustrate how to do this from the Eclipse editor.

**Screenshot: Right-click menu in Eclipse**

![Eclipse Right-click Menu with FishEye Options](image)

**RELATED TOPICS**

[Configuring your FishEye Options in Eclipse](#)
[More about Configuring FishEye Repositories in Eclipse](#)

## Using Crucible in the Eclipse Connector

The Atlassian Connector for Eclipse allows you to conduct [Crucible](#) reviews right there in your integrated development environment.

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the [Installation Guide](#).

- [Configuring your Crucible Options in Eclipse](#)
- [Working with Crucible Reviews in Eclipse](#)
Configuring your Crucible Options in Eclipse

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the Installation Guide.

On this page:
- Adding a Task Repository
- Setting Up a Crucible Task Repository

Adding a Task Repository

1. In Eclipse Mylyn, open the 'Task Repositories' view:
   - Click 'Window', 'Show View', 'Other'.
   - Open the 'Tasks' category and select the 'Task Repositories' view.
2. Click the 'Add Task Repository' icon.
3. The 'Add Task Repository' screen appears, as shown below:

   ![Add Task Repository](image)

4. Now you can select and configure one or more of the task repositories and configure them to match your server(s), as described below.
Once you have set up a repository, it will appear in the Eclipse 'Task Repositories' view, as shown in this screenshot:

Setting Up a Crucible Task Repository

- **Single server for both Crucible and FishEye = single task repository**

  If you have a single server for both FishEye and Crucible, you should set up the Crucible task repository and then select the 'Crucible Server Contains FishEye Instance' option as described below. Due to Mylyn requirements, you cannot set up two separate task repositories for the same URL.

1. On the Eclipse 'Add Task Repository' screen, select the 'Crucible' task repository type and click 'Next'.
2. The 'Crucible Repository Settings' screen appears, as shown below:

3. Enter the following information:
   - **Server** — The location (URL) of your Crucible server.
   - **Label** — A descriptive name for your Crucible server, e.g. 'Extranet Crucible'.
   - **Disconnected** — If necessary, you can tick this checkbox to disable a particular server without deleting it. This is useful if your servers are behind a firewall and you do not have access to them at a particular point in time.
   - **User ID** — The username you use to connect to your Crucible server.
   - **Password** — Your password on the Crucible server, matching the above username.
3. **Save Password** — Put a tick in the checkbox if you want to save your password on disk. Leave the checkbox unticked if you want to be asked for a password every time you start your IDE.

If you choose to save the password, it is stored on your computer in a file that is difficult, but not impossible, for an intruder to read.

4. Click the 'Validate Settings' button to verify the information you have entered.

5. Choose a **Review Activation** setting. These settings determine what happens when you open a source code file from within the Crucible review editor in Eclipse. For example, you may open a Crucible review within Eclipse. The review will open in an Eclipse editor view, and its 'Review Files' section will show the files in the review. You can then click a file to open it in an Eclipse editor. The options are as follows:
   - **'Always'** — The Crucible review will be automatically activated if you open a file from the Crucible review editor.
   - **'Never'** — The Crucible review will never be automatically activated.
   - **'Prompt'** (default) — When you open a file from the Crucible review editor, the connector will ask you whether you want to activate the Crucible review. This is the default setting.

6. If your Crucible server is linked to a FishEye server:
   - Put a tick in the checkbox labelled **'Crucible Server Contains FishEye Instance'**.
   - Set up your FishEye mappings, as described in the FishEye section of this documentation.

7. Click 'Finish' to save the changes.

8. Eclipse Mylyn will prompt you to add a new query for the new Crucible repository. This is where you will choose your Crucible filter, to determine which reviews appear in your task list. You can do this now or skip this step and do it later. You can also add more queries later.


You can configure one or more Crucible repositories, i.e. you can connect to more than one Crucible server.

### RELATED TOPICS

- Working with Crucible Reviews in Eclipse
- Installation and Upgrade Guide for the Eclipse Connector

### Working with Crucible Reviews in Eclipse

The Atlassian Connector for Eclipse allows you to conduct Crucible reviews right there in your integrated development environment. Your Crucible reviews will appear in your Eclipse 'Task List' view.

Before reading the information below, please make sure that you have installed the Atlassian Connector for Eclipse, as described in the installation guide, and defined a task repository for at least one Crucible server, as described in the configuration guide.

### Summary of What You Can Do

Below are the highlights of viewing and acting upon Crucible reviews within Eclipse. Follow the links to the relevant sections of the user guide.

- Add a query to retrieve and filter your reviews in Eclipse.
- View a list of your reviews in the Eclipse task list.
- Receive notification of new and updated reviews.
- View the review details in an Eclipse editor.
- Add, edit and reply to review comments in the review view or in the code editor.
- Open the file under review, at the commented source code line.
- Create a post-commit review, a patch review or a pre-commit review.
- Assign reviewers.
- Add changesets or patches to an existing review.
- Summarise and close the review and perform other workflow actions.
- Work on a review in the diff view.
- Work on a review in the review perspective.

### Adding a Query to Retrieve and Filter your Reviews

To retrieve your Crucible reviews and display them in your Eclipse task list, you will create an Eclipse Mylyn query. The query allows you to select an existing Crucible filter or define a custom filter.

When you define a task repository for your Crucible server, Eclipse Mylyn will prompt you to add a query for the new Crucible repository. You can also add more queries later, as described below.

To add a query for filtering your Crucible reviews,
1. If you have not already set up a task repository for your Crucible server, please do so by following the steps in the configuration guide.

2. Open your Eclipse 'Task Repositories' view and find the task repository that you defined for your Crucible server:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears, as shown in screenshot 1 below. Open the 'Tasks' folder and click 'Task Repositories'.
   - The 'Task Repositories' view appears, as shown in screenshot 2 below. Right-click the task repository that you defined for your Crucible server and select 'New Query'.

3. The 'Edit Repository Query' window appears, as shown in screenshot 3 below. Now you can choose to define a custom filter or use a predefined filter that exists on your Crucible server.
   - To define a custom filter:
     a. Select the 'Custom' radio button.
     b. Click 'Next'.
     c. Enter a title for your query, as prompted. Select the review field values that you want to filter on. For example, see screenshot 4 below.
     1. When selecting the review 'State', you can select more than one value by holding down the Ctrl key.
     d. Click 'Finish'.
   - To select an existing filter from your Crucible server:
     a. Select the 'Predefined Filter' radio button.
     b. Click one of the filters offered below the radio button, as shown in screenshot 3 below.
     c. Click 'Finish'.

4. Your Crucible reviews will appear in your 'Task List' view, as described below.

Screenshot 1: Opening the 'Task Repositories' view

![Show View](image1)

Use F2 to display the description for a selected view.

OK  Cancel

Screenshot 2: Adding a new query
Screenshot 3: Selecting a predefined review filter
Screenshot 4: Defining a custom review filter
Viewing your Reviews in the Eclipse Task List

Crucible reviews are included in your Eclipse 'Task List' view. You can take advantage of Mylyn's context management to keep track of the files and other assets associated with your review.

To open your Eclipse 'Task List' view,

1. In the Eclipse toolbar, select 'Window', 'Show View', 'Task List'.
2. The 'Task List' view appears, as shown in screenshot 5 below. Open the folder with the name corresponding to the query, i.e. the Crucible filter, that you created earlier.

Screenshot 5: Viewing your Crucible reviews in an Eclipse task list
Next, take a look at how you can monitor review updates in Eclipse.

**RELATED TOPICS**

- Installation and Upgrade Guide for the Eclipse Connector
- Configuring the Eclipse Connector
- Using JIRA in the Eclipse Connector
- Using Bamboo in the Eclipse Connector
- Using FishEye in the Eclipse Connector
- Using Crucible in the Eclipse Connector
- Eclipse Connector FAQ
- Eclipse Connector Resources

Refer to the Crucible documentation for more information about Crucible reviews.

**Monitoring Review Updates in Eclipse**

You will receive an Eclipse notification whenever one of the following occurs:

- Someone assigns you as a reviewer.
- There is a change to a review that is included in your Eclipse task list, retrieved via the queries that you have set up. (See the guide to creating a query.)

Here is an example of the Eclipse notification popup window.

*Screenshot: Crucible review notification*
Next, take a look at the Crucible review details, as displayed in an Eclipse editor.

RELATED TOPICS

- Working with Crucible Reviews in Eclipse
- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
- Managing Review Comments in Eclipse
- Working on a Review in the Diff View in Eclipse
- Creating a Post-Commit Review from within Eclipse
- Creating a Patch Review from within Eclipse
- Conducting a Pre-Commit Review from within Eclipse
- Assigning Reviewers in Eclipse
- Adding Changesets or Patches to Review in Eclipse
- Using the Crucible Review Perspective

Viewing a Review in Eclipse

Once you have set up your queries to retrieve and filter your Crucible reviews, you can open a review and see its details without leaving Eclipse. You can open a review from the Eclipse task list or from the code editor.

Tip: You can also view and work with a review in the Crucible Review Perspective.

To open a Crucible review from the Eclipse task list,

1. If you have not already done so, define your Crucible filters as described in the guide to creating a query.
2. Open your Eclipse ‘Task List’ view: In the Eclipse toolbar, select 'Window', 'Show View', 'Task List'.
3. Open the folder with the name corresponding to the query that you want, i.e. the Crucible filter that you created earlier.
4. Double-click the line showing the review you want, or right-click the line and select ‘Open’.
5. The review details appear in an Eclipse editor, as shown in the diagram below.

To open a Crucible review from the Eclipse code editor,

1. Make sure that the Crucible review task is active in Mylyn Eclipse. To activate the Mylyn task, right-click the review in your Eclipse Mylyn task list and select ‘Activate’.
2. Scroll down your code editor until you see a review comment marked by a Crucible icon in the left-hand ruler.
3. Hover over the Crucible icon to see the comment text and other details.
4. Press F2 to bring the comment into focus.
5. The review comment box becomes an inline dialogue box. Click ‘Open Review’ to open the review in the Eclipse review editor.

If you have trouble finding the Crucible comments in code editor, take a look at the full instructions with screenshots.

Diagram: Viewing the Crucible review details
The Crucible review details screen opens in an Eclipse editor. It displays the following:

- The review summary, state, date of opening and statement of objectives.
- The participants in the review, i.e. the author, moderator and reviewers.
- The files included in the review, and the general comments at file level for each file.
- Comments at line level within the file.
- Further options to:
  - Open the review in your browser, displaying Crucible’s web interface. See the diagram above.
  - Manage comments at review and file level: Add, reply to, edit and remove comments. See Managing Review Comments in Eclipse.
  - Open the file under review, at the commented source code line. See the diagram above.
  - Perform the workflow actions that are relevant to the review’s current state, such as summarise and close the review or abandon the review. See the diagram above.
  - Add reviewers to and remove reviewers from the review. See Assigning Reviewers in Eclipse.
  - Add changesets or a patch to the review. See Adding Changesets or Patches to Review in Eclipse.
  - Open a diff view and conduct your review in the diff. See Working on a Review in the Diff View in Eclipse.

Next, take a look at managing your review comments.

RELATED TOPICS

- Working with Crucible Reviews in Eclipse
- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
- Managing Review Comments in Eclipse
- Working on a Review in the Diff View in Eclipse
- Creating a Post-Commit Review from within Eclipse
- Creating a Patch Review from within Eclipse
- Conducting a Pre-Commit Review from within Eclipse
- Assigning Reviewers in Eclipse
Managing Review Comments in Eclipse

Once you have set up your Crucible server as an Eclipse task repository, you can perform a Crucible review without leaving Eclipse. The Atlassian Connector for Eclipse allows you to create and manage comments in a special Eclipse editor that contains the review, or directly in the Eclipse code editor. We describe both methods below. You can also manage your review comments in the diff view, as described in another topic.

Can't see or add comments in your code editor?
If you want to work with your Crucible review in the Eclipse code editor, the review task must be active in Mylyn Eclipse. See below for instructions.

Tip: Wiki markup is supported for review comments in the Atlassian Connector for Eclipse.

Tip: You can also view and work with review comments in the Crucible Review Perspective.

On this page:
- Managing Comments in the Review Editor
- Viewing Comments in the Code Editor
- Adding, Updating and Replying to Comments in the Code Editor

Managing Comments in the Review Editor

You can open your Crucible review in a special Eclipse editor and add, reply to or update review comments within the editor view.

To manage your comments in the review editor,

1. If you have not already done so, define your Crucible filters as described in the guide to creating a query.
2. Open the review in an Eclipse editor:
   - Open your Eclipse 'Task List' view: In the Eclipse toolbar, select 'Window', 'Show View', 'Task List'.
   - Open the folder with the name corresponding to the query that you want, i.e. the Crucible filter that you created earlier.
   - Double-click the line showing the review you want, or right-click the line and select 'Open'.
3. The review details appear in an Eclipse editor. Now you can conduct your review in the Eclipse editor, by adding and responding to review comments as shown in diagram 1 below:
   - Add, edit, delete and reply to comments applicable to the review as a whole. These are called 'general review comments'.
   - Add, edit, delete and reply to comments applicable to a file. These are called 'general file comments'.
   - Edit, delete and reply to comments applicable to a line or block of source code. These are called 'line comments'.

   To add a line comment, you need to open the source file in the Eclipse editor. See below. You cannot add line comments from the review details editor.
   - Open the file under review, at the commented source code line. This will open the file in the Eclipse code editor, so that you can manage your comments there as described below.

Working with code and review side by side
This is a handy way to work with the comments in the review editor while viewing the commented code in another open window: Open the source file in your code editor and the Crucible review in the review editor, with the two editor windows side by side. Click the text of a line comment in the review editor. The corresponding code lines will be highlighted in the code editor. You may find this easier than reading the code and hovering over the Crucible icon to see the comment.

Diagram 1: Managing your review comments in the review editor
Viewing Comments in the Code Editor

When working with your code in an Eclipse editor, you can see the Crucible review comments as well. See diagram 2 below.

- Review comments are annotated with the Crucible icon in the left-hand ruler of the Eclipse editor.
  - Hover over the Crucible icon to see the comment text and other details of the review comment.
  - Press F2 when viewing the comment details to bring the comment into focus, so that you can update or reply to the comment. See below for more details on how to conduct the review.
- The code lines to which the comment applies are highlighted in the editor body.
- An annotation at the top of the right-hand ruler shows, on hover, how many active review comments there are for the file.
- Annotations in the right-hand ruler show every line comment in the entire file, including the code sections that are not currently visible in your window.
  - Click an annotation to jump to the commented code line.
  - Hover over an annotation to see a tooltip containing the comment text.

Diagram 2: Viewing the review comments in the code editor
Adding, Updating and Replying to Comments in the Code Editor

When viewing the source code in an Eclipse editor, you can:

- Add general file comments.
- Add, reply to and update line comments.

To add comments in the code editor,

1. Make sure that the Crucible review task is active in Mylyn Eclipse. To activate the Mylyn task, you can do one of the following:
   - Open the review in its own Eclipse view, as described above.
   - Or right-click the review in your Eclipse Mylyn task list and select 'Activate'.
2. Select a code line or a block of lines in the Eclipse code editor and right-click.
3. Select either 'Add General File Comment' or 'Comment on Selected Lines'. In diagram 3 below, we show you how to add a line comment.
4. A popup window will ask you for the comment details. Enter the comment text, supply the other information if necessary, and 'Post' the comment.

Diagram 3: Adding review comments in the code editor
Review comments are annotated with the Crucible icon in the left-hand ruler of the Eclipse editor. Also, annotations in the right-hand ruler show every line comment in the entire file, including the code sections that are not currently visible in your window. See above for more details about viewing comments in the code editor.

To reply to and update line comments in the code editor,

1. Make sure that the Crucible review task is active in Mylyn Eclipse. To activate the Mylyn task, you can do one of the following:
   - Open the review in its own Eclipse view, as described above.
   - Or right-click the review in your Eclipse Mylyn task list and select ‘Activate’.
2. Find the comment that you want to reply to or update in the code editor:
   - Hover over the annotations in the right-hand ruler to see a tooltip containing the comment text.
   - Click the annotation in the right-hand ruler to jump to the comment you want.
3. Open the comment for editing or replying:
   - Hover over the Crucible icon in the left-hand ruler to see the comment text.
   - Press F2 when viewing the comment details to bring the comment into focus.
4. The review comment box becomes an inline dialogue box. Now you can reply to the comment, edit it or delete it, depending on your permissions, as shown in diagram 4 below. You can also choose to open the review in the Eclipse review editor instead. See above for instructions on managing comments in the review editor.

Diagram 4: Replying to and updating line comments in the code editor
Next, take a look at working on a review in the diff view.

RELATED TOPICS

Working with Crucible Reviews in Eclipse

- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
- Managing Review Comments in Eclipse
- Working on a Review in the Diff View in Eclipse
- Creating a Post-Commit Review from within Eclipse
- Creating a Patch Review from within Eclipse
- Conducting a Pre-Commit Review from within Eclipse
- Assigning Reviewers in Eclipse
- Adding Changesets or Patches to Review in Eclipse
- Using the Crucible Review Perspective

Refer to the Crucible documentation for more information about Crucible reviews.

Working on a Review in the Diff View in Eclipse

Once you have set up your Crucible server as an Eclipse task repository, you can perform a Crucible review without leaving Eclipse. The Atlassian Connector for Eclipse allows you to create and manage comments in your Eclipse code editor, as described in another topic. You can also manage your review comments in the diff view, as described below.

Can't see comments in your code editor?
If you want to work with your Crucible review in the Eclipse code editor, the review task must be active in Mylyn Eclipse. See the instructions on managing review comments.

To add review comments in a diff view,
1. Open the Crucible review in an Eclipse review editor. Follow these instructions to open the review from your code editor, or see the complete instructions on opening a review:
   - Make sure that the Crucible review task is active in Mylyn Eclipse. To activate the Mylyn task, right-click the review in your Eclipse Mylyn task list and select 'Activate'.
   - Scroll down your code editor until you see a review comment marked by a Crucible icon in the left-hand ruler.
   - Hover over the Crucible icon to see the comment text and other details.
   - Press F2 to bring the comment into focus.
   - The review comment box becomes an inline dialogue box. Click 'Open Review' to open the review in the Eclipse review editor.

2. Open the diff view by clicking 'Compare' next to the file you want, in the 'Review Files' section of the review editor as shown in diagram 1 below.

3. The 'Compare' view opens, as shown in diagram 1 below. Select a code line or a block of lines in the Eclipse code editor and right-click.

4. Select either 'Create General File Comment' or 'Create Line Comment'. In diagram 1 below, we show you how to add a line comment.

5. A popup window will ask you for the comment details. Enter the comment text, supply the other information if necessary, and 'Post' the comment.

---

Diagram 1: Adding review comments in a diff view

To reply to and update review comments in a diff view,
1. Open the Crucible review in an Eclipse review editor. Follow these instructions to open the review from your code editor, or see the complete instructions on opening a review:
   - Make sure that the Crucible review task is active in Mylyn Eclipse. To activate the Mylyn task, right-click the review in your Eclipse Mylyn task list and select ‘Activate’.
   - Scroll down your code editor until you see a review comment marked by a Crucible icon in the left-hand ruler.
   - Hover over the Crucible icon to see the comment text and other details.
   - Press F2 to bring the comment into focus.
   - The review comment box becomes an inline dialogue box. Click ‘Open Review’ to open the review in the Eclipse review editor.

2. Open the diff view by clicking ‘Compare’ next to the file you want, in the ‘Review Files’ section of the review editor as shown in diagram 2 below.
3. The ‘Compare’ view opens, as shown in diagram 2 below. Scroll down until you see a review comment marked by a Crucible icon in the left-hand ruler.
   - Hover over the Crucible icon to see the comment text and other details.
   - Press F2 to bring the comment into focus.
   - The review comment box becomes an inline dialogue box. Now you can reply to the comment, edit it or delete it, depending on your permissions.

Diagram 2: Working with review comments in a diff view

Next, take a look at creating a post-commit review.

RELATED TOPICS

Working with Crucible Reviews in Eclipse
- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
Creating a Post-Commit Review from within Eclipse

Once you have set up your Crucible server as an Eclipse task repository, you can create a Crucible review without leaving Eclipse. The Atlassian Connector for Eclipse allows you to create post-commit reviews, as described below, and also patch reviews, as described in another topic.

A post-commit review allows you to ask your colleagues to review changesets that have already been committed to your source repository. You can create a post-commit review by creating a new task via the task repository, or by from files selected in the package explorer, or from the Java file displayed in the Active Task Editor.

On this page:
- Creating a Post-Commit Review from the Task Repository
- Creating a Post-Commit Review from the Package Explorer
- Creating a Post-Commit Review from a file in the Active Task Editor

Creating a Post-Commit Review from the Task Repository

To create a post-commit review from the task repository,

1. If you have not already set up a task repository for your Crucible server, please do so by following the steps in the configuration guide.
2. Open your Eclipse 'Task Repositories' view and find the task repository that you defined for your Crucible server:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears. Open the 'Tasks' folder and click 'Task Repositories'.
   - The 'Task Repositories' view appears, as shown in diagram 1 below. (If you have trouble finding the 'Task Repositories' view, try the full instructions with screenshots.)
3. Right-click the task repository that you defined for your Crucible server and select 'New Task'.
4. The 'New Crucible Review' window appears, as shown in diagram 1 below. Select 'From a Changeset' and click 'Next'.
5. The 'New Crucible Review' window now shows recent changesets from your repository, as shown in screenshot 2 below. Select one or more changesets from the list on the left. (Hold down the Ctrl key to select more than one changeset at once.)
6. Click 'Add' to move the selected changeset(s) into the box on the right.
7. If prompted, click the 'Edit' button to map your local repository to the repository on your Crucible server.
8. Click 'Next'.
9. The 'New Crucible Review' window now prompts you for the review title and other details, as shown in screenshot 3 below. Enter the details and click 'Finish'.

Diagram 1: Adding a review as a task in your Crucible repository
Screenshot 2: Adding changesets to your review

Screenshot 3: Adding the review details

Add a post-commit review
Next, take a look at creating a patch review.

Creating a Post-Commit Review from the Package Explorer

To create a post-commit review from the package explorer,
1. Open your Eclipse 'Package Explorer'. You can view this in the 'Java' perspective, if you are currently using a different perspective.
2. Locate the files to be included in the review, in the file tree in the Package Explorer and select them.
3. Right-click one of the selected files and select 'Team' -> 'Create Review' from the popup menus. The 'Select a repository' window will display.
4. Select the repository that you are reviewing files from and click 'Next'. The 'Add Resources to Review' window will display with the files you selected.
   
   If you have selected unversioned or modified resources, they will be added as pre-commit review items.

5. Click the 'Next' button. The 'New Crucible Review' window will display (as per Screenshot 3 above).
6. Enter the review details as desired and click the 'Finish' button to create your review.

Creating a Post-Commit Review from a file in the Active Task Editor

To create a post-commit review from a file in the Active Task Editor,

1. Open a Java file in the Active Task Editor.
2. Right-click the file in the Active Task Editor and select 'Team' -> 'Create Review' from the popup menus. The 'Select a repository' window will display.
3. Select the repository that you are reviewing files from and click 'Next'. The 'New Crucible Review' window will display (as per Screenshot 3 above).
4. Enter the review details as desired and click the 'Finish' button to create your review.

RELATED TOPICS

- Working with Crucible Reviews in Eclipse
- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
- Managing Review Comments in Eclipse
- Working on a Review in the Diff View in Eclipse
- Creating a Post-Commit Review from within Eclipse
- Creating a Patch Review from within Eclipse
- Conducting a Pre-Commit Review from within Eclipse
- Assigning Reviewers in Eclipse
- Adding Changesets or Patches to Review in Eclipse
- Using the Crucible Review Perspective

Refer to the Crucible documentation for more information about Crucible reviews.
Creating a Patch Review from within Eclipse

Once you have set up your Crucible server as an Eclipse task repository, you can create a Crucible review without leaving Eclipse. The Atlassian Connector for Eclipse allows you to create patch reviews, as described below, and also post-commit reviews, as described in another topic.

A patch review allows you to ask your colleagues to review a proposed change before you have committed it to your source repository.

To create a patch review,

1. Create the patch from the changes you have made to your source file in the Eclipse editor, and copy the patch to your clipboard. Here are some brief instructions, in case you have not done this before:
   - Save the changes you have made to your source file locally. There is no need to commit these changes to your source repository.
   - Right-click on your source file in the Eclipse editor and select 'Team', 'Create Patch'.
   - The 'Create Patch' window appears. Select 'Save to Clipboard'.
   - Click 'Next'.
   - Leave the 'Advanced options' set to the default values. Click 'Finish'.
2. If you have not already set up a task repository for your Crucible server, please do so by following the steps in the configuration guide.
3. Open your Eclipse 'Task Repositories' view and find the task repository that you defined for your Crucible server:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears. Open the 'Tasks' folder and click 'Task Repositories'.
   - The 'Task Repositories' view appears, as shown in diagram 1 below. (If you have trouble finding the 'Task Repositories' view, try the full instructions with screenshots.)
4. Right-click the task repository that you defined for your Crucible server and select 'New Task'.
5. The 'New Crucible Review' window appears, as shown in diagram 1 below. Select 'From a Patch' and click 'Next'.
6. The 'New Crucible Review' window now prompts you to add your patch from your clipboard, as shown in screenshot 2 below. Your patched change will appear in the box on the page, because you have already copied it to your clipboard.
7. Select 'Include this Patch from the clipboard to the review'.
8. Select your repository on your Crucible server.
9. Click 'Next'.
10. The 'New Crucible Review' window now prompts you for the review title and other details, as shown in screenshot 3 below. Enter the details and click 'Finish'.

Diagram 1: Adding a review as a task in your Crucible repository
Screenshot 2: Adding a patch to your review
Screenshot 3: Adding the review details
Next, see how to conduct a pre-commit review.

RELATED TOPICS

- Working with Crucible Reviews in Eclipse
- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
- Managing Review Comments in Eclipse
- Working on a Review in the Diff View in Eclipse
- Creating a Post-Commit Review from within Eclipse
- Creating a Patch Review from within Eclipse
- Conducting a Pre-Commit Review from within Eclipse
- Assigning Reviewers in Eclipse
- Adding Changesets or Patches to Review in Eclipse
- Using the Crucible Review Perspective

Refer to the Crucible documentation for more information about Crucible reviews.

Conducting a Pre-Commit Review from within Eclipse

Once you have set up your Crucible server as an Eclipse task repository, you can create a pre-commit Crucible review from within Eclipse. This allows you to ask your team members to review your code before you commit it to the repository. The review gathers your local working copies of the files. You and your reviewers can then open the changed files for comment in Eclipse, even though the files have not been committed to the source repository.

Currently the connector supports pre-commit reviews with Subclipse and Subversive only. You cannot yet do pre-commit reviews using CVS.

On this page:
- Creating a Pre-Commit Review
- Performing a Pre-Commit Review

Creating a Pre-Commit Review

The easiest way to create a pre-commit review is from the 'Team' option in the context menu that appears when you right-click an explorer or
To create a pre-commit review from a context menu,

1. Right-click a line in the Package Explorer or Eclipse Editor window, for example. Then select 'Team' and 'Create Review', as shown in screenshot 1 below. The 'Select a repository' window will display.
2. Select your Crucible repository and click 'Next', as shown in screenshot 2 below. The 'Add Resources to Review' window will display.
   - If you have not set up your repository, click the 'Add Task Repository' button to do so.
3. Select the changed files from your workspace, as shown in screenshot 3 below.
   - If you haven't set up a repository mapping as described in the FishEye configuration guide, you will need to do so before you can continue.
4. Enter the review title and other details as shown in screenshot 4 below, then click 'Finish'.

Screenshot 1: Creating a pre-commit review from a context menu

![Screenshot 1: Creating a pre-commit review from a context menu](image1.png)

Screenshot 2: Selecting the repository

![Screenshot 2: Selecting the repository](image2.png)
Screenshot 3: Selecting the changes for review

New Crucible Review
Add selected resources to Review. Modified or unversioned resources will be added as pre-commit review items.

Include resources:
- test23/src/test/java/com/atlassian/upgrade
  - DummyDatabaseUpgradeTask.java
  - DummyUpgradeState.java
  - DummyUpgradeTask.java

Define Repository Mappings
To create a pre-commit review from your task repository,

1. If you have not already set up a task repository for your Crucible server, please do so by following the steps in the configuration guide.
2. Open your Eclipse 'Task Repositories' view and find the task repository that you defined for your Crucible server:
   - In the Eclipse toolbar, select 'Window', 'Show View', 'Other'.
   - The 'Show View' popup window appears. Open the 'Tasks' folder and click 'Task Repositories'.
   - The 'Task Repositories' view appears, as shown in screenshot 4 below. (If you have trouble finding the 'Task Repositories' view, try the full instructions with screenshots.)
3. Right-click the task repository that you defined for your Crucible server and select 'New Task'.
4. The 'New Crucible Review' window appears, as shown in screenshot 5 below. Select 'From Workspace Changes' and click 'Next'.
5. The 'New Crucible Review' window now prompts you to add your workspace changes to the review, as shown in screenshot 6 below. Select the files that you want your colleagues to review and click 'Next'.
   Hint: Right-click on the list of changes to select or deselect all.
6. The 'New Crucible Review' window now prompts you for the review title and other details, as shown in screenshot 7 below. Enter the details and click 'Finish'.

Screenshot 4: Adding a review as a task in your Crucible repository

Screenshot 5: Selecting the review type
Screenshot 6: Selecting the files to be reviewed

Screenshot 7: Adding the review details
Performing a Pre-Commit Review

If you are invited to take part in a pre-commit review, you will open the review in Eclipse as described in Viewing a Review in Eclipse. The pre-commit review will look something like this:

```
Perform the available workflow actions. Shown here:
- Summarise and close review
- Abandon review
```

The Crucible review details screen opens in an Eclipse editor. It displays the following:

- The review summary, state, date of opening and statement of objectives.
- The participants in the review, i.e. the author, moderator and reviewers.
- The files included in the review, and the general comments at file level for each file.
- Comments at line level within the file.
- Further options to:
  - Open the review in your browser, displaying Crucible’s web interface.
  - Manage comments at review and file level: Add, reply to, edit and remove comments.
  - Open the old (previous) version of the file under review, showing the file contents before the updates were made.
Open the new (updated) version of the file under review.
Open a diff view of the file, comparing the old and new versions. See Working on a Review in the Diff View in Eclipse.
Perform the workflow actions that are relevant to the review's current state, such as summarise and close the review or abandon the review.
Add reviewers to and remove reviewers from the review. See Assigning Reviewers in Eclipse.
Add changesets or a patch to the review. See Adding Changesets or Patches to Review in Eclipse.

Next, take a look at assigning reviewers.

RELATED TOPICS

Working with Crucible Reviews in Eclipse
- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
- Managing Review Comments in Eclipse
- Working on a Review in the Diff View in Eclipse
- Creating a Post-Commit Review from within Eclipse
- Creating a Patch Review from within Eclipse
- Conducting a Pre-Commit Review from within Eclipse
- Assigning Reviewers in Eclipse
- Adding Changesets or Patches to Review in Eclipse
- Using the Crucible Review Perspective

Refer to the Crucible documentation for more information about Crucible reviews.

Assigning Reviewers in Eclipse

Once you have set up your Crucible server as an Eclipse task repository, you can add reviewers to or remove reviewers from a Crucible review without leaving Eclipse.

To assign reviewers in the Eclipse review editor,

1. If you have not already done so, define your Crucible filters as described in the guide to creating a query.
2. Open the review in an Eclipse editor:
   - Open your Eclipse 'Task List' view: In the Eclipse toolbar, select 'Window', 'Show View', 'Task List'.
   - Open the folder with the name corresponding to the query that you want, i.e. the Crucible filter that you created earlier.
   - Double-click the line showing the review you want, or right-click the line and select 'Open'.
3. The review details appear in an Eclipse editor, as shown in the diagram below. Click the 'Add/Remove Reviewers' icon in the 'Participants' section.
4. The 'Select Reviewer(s)' screen appears, showing a list of Crucible users available for the review. Narrow down the list of users by entering part of a name in the text box at the top.
   - To add a reviewer to the review, select the checkbox next to their name.
   - To remove a reviewer from the review, deselect the checkbox next to their name.

Diagram: Adding and removing reviewers in Eclipse
Next, take a look at adding changesets or patches to a review.

**RELATED TOPICS**

- Working with Crucible Reviews in Eclipse
- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
- Managing Review Comments in Eclipse
- Working on a Review in the Diff View in Eclipse
- Creating a Post-Commit Review from within Eclipse
- Creating a Patch Review from within Eclipse
- Conducting a Pre-Commit Review from within Eclipse
- Assigning Reviewers in Eclipse
- Adding Changesets or Patches to Review in Eclipse
- Using the Crucible Review Perspective

Refer to the [Crucible documentation](#) for more information about Crucible reviews.

**Adding Changesets or Patches to Review in Eclipse**

Once you have set up your Crucible server as an Eclipse task repository, you can add changesets, patches and files to a Crucible review without leaving Eclipse.

To add a changeset to a Crucible review,
1. If you have not already done so, define your Crucible filters as described in the guide to creating a query.

2. Open the review in an Eclipse editor:
   - Open your Eclipse 'Task List' view: In the Eclipse toolbar, select 'Window', 'Show View', 'Task List'.
   - Open the folder with the name corresponding to the query that you want, i.e. the Crucible filter that you created earlier.
   - Double-click the line showing the review you want, or right-click the line and select 'Open'.

3. The review details appear in an Eclipse editor, as shown in diagram 1 below. Click the 'Add Changesets' icon in the 'Review Files' section.

4. The 'Add Changeset' screen appears, showing a list of committed changesets in your repository, as shown in screenshot 2 below. Select one or more changesets from the list on the left. (Hold down the Ctrl key to select more than one changeset at once.)

5. Click 'Add' to move the selected changeset(s) into the box on the right.

6. Click the 'Edit' button to map your local repository to the repository on your Crucible server.

7. Click 'Finish'.

---

To add a patch to a Crucible review,

1. Create the patch from the changes you have made to your source file in the Eclipse editor, and copy the patch to your clipboard. Here are some brief instructions, in case you have not done this before:
   - Save the changes you have made to your source file locally. There is no need to commit these changes to your source repository.
   - Right-click on your source file in the Eclipse editor and select 'Team', 'Create Patch'.
   - The 'Create Patch' window appears. Select 'Save to Clipboard'.
   - Click 'Next'.
   - Leave the 'Advanced options' set to the default values. Click 'Finish'.

2. If you have not already done so, define your Crucible filters as described in the guide to creating a query.

3. Open the review in an Eclipse editor:
   - Open your Eclipse 'Task List' view: In the Eclipse toolbar, select 'Window', 'Show View', 'Task List'.
   - Open the folder with the name corresponding to the query that you want, i.e. the Crucible filter that you created earlier.
   - Double-click the line showing the review you want, or right-click the line and select 'Open'.

4. The review details appear in an Eclipse editor, as shown in diagram 1 below. Click the 'Add Patch' icon in the 'Review Files' section.

5. The 'Add Patch' window appears, as shown in screenshot 3 below. Your patched change will appear in the box on the page, because you have already copied it to your clipboard. Select 'Include this Patch from the clipboard to the review'.

6. Select your repository on your Crucible server.

7. Click 'Finish'.

---

To add files to a Crucible review,

1. Set the review that you wish to add files to, as the active review. You can do this via the Crucible Review Perspective.

2. Select the files that you wish to add to the active review in the 'Package Explorer' (Java perspective)/'Project Explorer' (Crucible Review Perspective), as shown in screenshot 4 below.

3. Right-click on the selected files and select 'Team' -> 'Add to Active Review'. The 'Add Resources to Review' window will display.

4. Confirm that you wish to add the files to the active review by clicking the 'Finish' button. The files will be added to your active review.

---

Diagram 1: Adding changesets and patches to a review
Select Changesets

Select the changesets that should be included in the review.

Select changesets from your repositories:

<table>
<thead>
<tr>
<th>Add</th>
<th>Remove</th>
</tr>
</thead>
</table>

Changesets selected for the review:

- https://studio.atlassian.com/svn/PLE/trunk
- 41093 [pniewiadomska] - PLE-583: fixed usage coll

Local Repository | Crucible Repository
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="https://studio.atlassian.com/svn/PLE/trunk">https://studio.atlassian.com/svn/PLE/trunk</a></td>
<td>PLE</td>
</tr>
</tbody>
</table>

Update Repository Data

Finish | Cancel
RELATED TOPICS

Working with Crucible Reviews in Eclipse

- Monitoring Review Updates in Eclipse
- Viewing a Review in Eclipse
- Managing Review Comments in Eclipse
- Working on a Review in the Diff View in Eclipse
- Creating a Post-Commit Review from within Eclipse
- Creating a Patch Review from within Eclipse
- Conducting a Pre-Commit Review from within Eclipse
- Assigning Reviewers in Eclipse
- Adding Changesets or Patches to Review in Eclipse
- Using the Crucible Review Perspective

Refer to the Crunchible documentation for more information about Crucible reviews.

Using the Crucible Review Perspective

A perspective in Eclipse is a pre-configured workspace layout. The connector includes a Crucible Review perspective that provides easy access to common review functions.

On this page:

- Working with the Crucible Review Perspective
- Viewing Comments in the Crucible Review Perspective
- Working with Comments in the Crucible Review Perspective

Working with the Crucible Review Perspective

To open the Crucible perspective,
1. Click the ‘Window’ menu, then select ‘Open Perspective’, ‘Other’.
2. Select ‘Crucible Review’ from the ‘Open Perspective’ window and click the ‘OK’ button.
3. Your workspace will be re-organised as shown in the screenshot below. You can switch to another perspective via the buttons in the top right of your IDE.
4. To set a review as the active review, click the Task List tab and click the ✅ icon next to the review, i.e.

   ![Review Explorer Screenshot]

   The review details will display in the main window.
5. Click the ‘Review Explorer’ tab to view information about and work with the active review.
   - To expand all folders, click the 📁 icon.
   - To collapse all folders, click the 📁 icon.
   - To link the review with the editor and comment view (i.e. the Review Explorer and Review Comment view will automatically jump to the file/comment that is currently in focus in the main editor window), click the 📁 icon.
   - To focus on unread comments only, click the 📁 icon. Comments which have been read will be greyed out and will be skipped when using the previous comment and next comment buttons, e.g.

   ![General Comments]

   - To collapse all folders, click the 📁 icon.
   - To navigate to the next comment, click the ⬇️ icon.
   - To navigate to the previous comment, click the ⬆️ icon.
   - To open the Active Task Editor (i.e. main window) if closed, click the ✅ icon.
   - To open the Comment View, if closed, click the ✅ icon.
   - To add a general comment to the review, click the ✉️ icon.
   - To publish all draft comments, click the ✍️ icon.
   - To add a comment to a particular file, click the file then click the ✍️ icon.

   **Viewing Comments in the Crucible Review Perspective**

   To view a general or file comment in the Crucible Review Perspective,
1. Open the Crucible Review Perspective as described above.

2. View the comments for the review, as follows:
   - To view the list of general comments for the review, expand the ’General Comments’ folder. Click a comment and the details of the comment will display in the Comment View (i.e. ’Review Comment’ tab in the bottom pane).

   ![General Comments]

   - I'm waiting for a decision on 'support'
   - I have reviewed the email texts and sug

   - To view the file-specific comments, first expand the ’Files’ folder, the files involved in the review will be shown. Expand the file in the list and you will be able to view the list of review comments against the specific file. Click a comment and the details of the comment will display in the Comment View (i.e. ’Review Comment’ tab in the bottom pane).

   ![Files]

   - support/supportsystem/jelly
   - support-closersolved-hosted.xml
   - support-closersolved.xml 2
   - Do we guarantee that the full
   - Good point. The full nam
   - support-inactivate-ehsp.xml
   - support-inactivate.xml

**Working with Comments in the Crucible Review Perspective**

To work with a comment in the Crucible Review Perspective,

1. Open the Crucible Review Perspective and then select a comment for display in the Comment View, as described above. The details of the comment will display, as seen in the following screenshot:

   ![Review Comment]

   - To reply to the comment, click the icon.
   - To edit the comment (if you are the author), click the icon.
   - To delete the comment (if you are the author), click the icon.
   - To publish the comment (if a draft), click the icon.
   - To leave the comment as unread, click the icon and select ‘Leave Unread’ from the dropdown menu.
   - To open the Active Task Editor (i.e. main window) if closed, click the icon.

**Screenshot: Crucible Review Perspective**
Eclipse Connector FAQ

Atlassian Connector for Eclipse — FAQ

Known issues, hints and tips and answers to commonly raised questions about the Atlassian Connector for Eclipse:

Configuration FAQ for the Eclipse Connector

- Using NTLM Authentication with the Eclipse Connector

Usage FAQ for the Eclipse Connector

- Custom Fields Do Not Work As Expected
- Customised Priorities Do Not Work
- Error Parsing Server Response - No Project Found
- Estimated Time Displayed in the Task Editor Differs from the Web Interface
- Invalid Date Error When Submitting a JIRA Issue
- Issue Dependencies Do Not Display in Task List
- Missing Attachments in Task Editor
- Unexpected Redirect Errors when Submitting an Issue

Installation FAQ for the Eclipse Connector

- Error while Collecting Items to be Installed
- Installation Error - Cannot Find a Solution Satisfying the Following Requirements

Configuration FAQ for the Eclipse Connector

Here is a list of all entries in the configuration FAQ, plus the first few lines of content. Click a link to see the full text for each entry.

- Using NTLM Authentication with the Eclipse Connector — NTLM support is limited by the libraries that Mylyn uses for network communication. A suggestion is to take a look at the NTLM authorisation proxy, which has been reported to work with the Mylyn JIRA connector.

Using NTLM Authentication with the Eclipse Connector

NTLM support is limited by the libraries that Mylyn uses for network communication. A suggestion is to take a look at the NTLM authorisation proxy, which has been reported to work with the Mylyn JIRA connector.

Usage FAQ for the Eclipse Connector

Here is a list of all entries in the usage FAQ, plus the first few lines of content. Click a link to see the full text for each entry.
Custom Fields Do Not Work As Expected

Custom fields are supported for the Atlassian Connector for Eclipse. However, the following limitations apply:

- Custom fields are not displayed in the task editor when creating new issues. This can cause issue creation to fail if custom fields are mandatory.
- Custom fields that are specific to a workflow state, e.g. are only displayed when resolving an issue, are not displayed in the task editor. This can cause a workflow transition to fail if custom fields are mandatory.
- If a value is not set for a custom field it will not be displayed in the task editor.
- Single-select and multi-select fields are not editable and options for these fields are not displayed in the task editor.

See this issue for additional information.

Customised Priorities Do Not Work

Customised priorities are not currently supported, so all issues with a customised priority will have the default priority level.

Please see this issue for further information.

Error Parsing Server Response - No Project Found

Symptoms

Error parsing server response: No project with key ‘xyz’ found

Cause

This error is caused by Mylyn not having an up-to-date copy of the repository configuration for that repository.

Resolution

Right-click on the query and select Repository -> Update Attributes. That will download and store a fresh copy of the repository configuration.

Estimated Time Displayed in the Task Editor Differs from the Web Interface

Mylyn defaults to a day having 24 working hours and a week having 7 work days. If your JIRA repository uses different settings, then time tracking information will not show correctly when converted between Mylyn and JIRA.

To fix this, change the time tracking settings in Mylyn in the Additional Settings section of the repository properties (Window -> Show View -> Task Repositories) to match the server configuration. Your JIRA administrator will be able to tell you the correct settings which can be located under Administration -> Time Tracking in the JIRA web interface.

Screenshot: JIRA Web Interface

<table>
<thead>
<tr>
<th>FIND ISSUES</th>
<th>CREATE NEW ISSUE</th>
<th>ADMINISTRATION</th>
</tr>
</thead>
</table>
| Time Tracking

Time Tracking is currently ON.
The number of working hours per day is 8.
The number of working days per week is 5.
Time estimates will be displayed in the following format: pretty (e.g. 4 days, 4 hours, 30 minutes)

Note: To change these values deactivate and then reactivate Time Tracking.

Screenshot: Mylyn Task Repository Settings
Invalid Date Error When Submitting a JIRA Issue

Symptom
Error message "You did not enter a valid date" displays when submitting a JIRA issue.

Cause
Your server may be configured to use a custom date format or a non English locale.

Resolution
You can change the date format settings in the 'Advanced Configuration' section under the 'Additional Settings' section of the repository properties (Window -> Show View -> Task Repositories) to match your server configuration.

Issue Dependencies Do Not Display in Task List

Symptom
Issue dependencies are not visible in the Task List.

Cause
By default, only subtask relationships are displayed in the Task List.

Resolution
You can enable 'Show linked tasks' in the 'Additional Settings' section of the repository properties (Window -> Show View -> Task Repositories) to also show issue links in the Task List. After enabling the setting 'Synchronize All Tasks' from the popup menu of the task repository, the Task Repositories view needs to be selected to update all tasks.

See this issue for further information.

Missing Attachments in Task Editor

Attachment retrieval including retrieval of task contexts and attachment lists in the task editor requires JIRA 3.7 or above. Please upgrade your JIRA installation to resolve this issue.

See this issue for more information.

Unexpected Redirect Errors when Submitting an Issue

Please go to the repository properties and validate the repository. If errors about circular redirects are logged in your Eclipse error log, your JIRA server or proxy may not be configured correctly.

In some cases unexpected redirects can be caused by a JIRA server accessed via Apache and mod_jk (using the AJP/1.3 Tomcat connector). Some server configurations may benefit from the 'proxyHost', 'proxyPort', 'scheme' and 'secure' settings on the AJP Connector in
the Tomcat server configuration.

Please see this issue for more information.

**Installation FAQ for the Eclipse Connector**

Here is a list of all entries in the installation troubleshooting section. Click a link to see the full text for each entry.

- Error while Collecting Items to be Installed
- Installation Error - Cannot Find a Solution Satisfying the Following Requirements

**Error while Collecting Items to be Installed**

**Symptoms**

During installation, after the license agreement screen, you may receive an error message similar to this:

```
An error occurred while collecting items to be installed
No repository found containing: com.atlassian.connector.commons/osgi.bundle/0.5.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.bamboo.core/osgi.bundle/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.bamboo/org.eclipse.update.feature/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.bamboo.ui/osgi.bundle/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.crucible.core/osgi.bundle/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.crucible.org.eclipse.update.feature/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.crucible.ui/osgi.bundle/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.core/osgi.bundle/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.crm.core/osgi.bundle/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.crm.org.eclipse.update.feature/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.crm.ui/osgi.bundle/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.core/org.eclipse.update.feature/1.0.0.I20090320-BETA
No repository found containing: com.atlassian.connector.eclipse.crm/org.eclipse.update.feature/1.0.0.I20090320-BETA
No repository found containing: com.thoughtworks.xstream/osgi.bundle/1.3.0.v20090224-1640
No repository found containing: org.apache.commons.io/osgi.bundle/1.3.2.v20080604-1500
No repository found containing: org.jdom_jaxen/osgi.bundle/1.0.0.20081203-1100
No repository found containing: org.joda.time/osgi.bundle/1.6.0.v20081202-0100
```

**Cause**

P2 (Eclipse's update mechanism in Eclipse 3.4 and higher) has cached some incorrect information about the update site where the plugins reside for the Atlassian Connector for Eclipse.

**Resolution**

Follow these steps to clear the cached information:

1. Remove the Atlassian update site:

   ![Image](http://update.atlassian.com/atlassian-eclipse-plugin/e3.4)

2. Add the Atlassian update site again:

   ![Image](http://update.atlassian.com/atlassian-eclipse-plugin/e3.4)

3. Try to install the Atlassian Connector for Eclipse again.

**More Information**

See issue PLE-393 in our bug tracker.

**RELATED PAGES**

**Installing the Eclipse Connector**

**Installation Error - Cannot Find a Solution Satisfying the Following Requirements**

**Symptoms**

During installation you may receive an error message similar to one of these:

- Cannot complete the request. See the details.
  Cannot find a solution satisfying the following requirements Match[requiredCapability: org.eclipse.equinox.p2.ui/org.eclipse.swt/[3.4.0.v3448f,3.4.0.v3448f]]

- Cannot complete the request. See the details.
  Cannot find a solution where both Match[requiredCapability: org.eclipse.equinox.p2.ui/org.eclipse.swt/[3.4.1.v3449c,3.4.1.v3449c] and Match[requiredCapability: org.eclipse.equinox.p2.ui/org.eclipse.swt/[3.4.0.v3448f,3.4.0.v3448f] can be satisfied.
  Unsatisfied dependency: [org.eclipse.mylyn.commons.soap 1.0.0.I20090109-0100-e3x] requiredCapability: osgi.bundle/org.eclipse.mylyn.commons.net/[3.1.0,4.0.0)

- Cannot complete the request. See the details.
Cannot find a solution satisfying the following requirements org.eclipse.pde.feature.jar
[3.4.0.v20080603-777U_E_EkMNnKnb39WcMC].

- Cannot complete the install because one or more required items could not be found.
  Software being installed: Atlassian Connector for Eclipse (recommended) 1.1.0.v20090624
  (com.atlassian.connector.eclipse.feature.group 1.1.0.v20090624)
  Missing requirement: Atlassian Connector for Eclipse (recommended) 1.1.0.v20090624
  (com.atlassian.connector.eclipse.feature.group 1.1.0.v20090624) requires
  'org.eclipse.mylyn.jira_feature.feature.group [3.2.0.I20090529,3.3.0]' but it could not be found

**Cause**

This is most likely an error from P2 (Eclipse's update mechanism in Eclipse 3.4 and higher). The problem seems to be that the Mylyn update site is not enabled in your copy of Eclipse. There is a bug in Eclipse that these sites may not be automatically enabled even though we specify that we require them for installation.

If any of the installed features have unsatisfied dependencies or if features were previously installed from the extras or incubator update site, P2 may fail with an error similar to the one above.

**Resolution**

Follow these steps to enable the required sites:

1. In Eclipse, open 'Help', 'Software Updates'.
2. Click the 'Available Software' tab.
3. Click 'Manage Sites'.
4. Look for either the 'Mylyn Extras' update site or a URL that looks like this:
   ```
   ```
5. Make sure that the site or URL is checked.
6. Look for either the 'Mylyn For Eclipse 3.4' update site or a URL that looks like this:
   ```
   1.http://download.eclipse.org/tools/mylyn/update/e3.4
   ```
7. Make sure that the site or URL is checked.
8. Now you should update all installed features to the latest version. Click the 'Installed Software' tab.
9. Click 'Update'.
10. Restart if prompted.
11. Try to install the Atlassian Connector for Eclipse again.

**More Information**

- Issue PLE-576.
- Issue PLE-393.
- Issue PLE-506.
- The Mylyn FAQ.

**Illustration**

Here is a screenshot showing an example of one such error message:
Eclipse Connector Resources

The Atlassian Connector for Eclipse is an Eclipse plugin. It allows you to work with the Atlassian products within your Eclipse IDE. Now you don't have to switch between websites, email messages and news feeds to see what's happening to your project and your code. Instead, you can see the relevant JIRA issues, Crucible reviews and Bamboo build information right there in your development environment.

Resources for Evaluators

- Install for free
- Feature tour

Downloadable Documentation

- Connector documentation in PDF, HTML or XML formats

Support

- Atlassian Support

Forums

- Atlassian IDE Connector Forum

Feature Requests

- Issue tracker and feature requests

Source

- This is an open source project. To access the connector's source code, point your SVN at: https://studio.atlassian.com/svn/PLE/trunk.
- If you don't already have an account, please go to http://studio.atlassian.com and sign up for an account. You should then be able to access the svn repository with your new account.

Atlassian Integration
Guide to Installing an Atlassian Integrated Suite

**Looking for Other Plugins?**

If you're in the wrong place, try these links instead:

- Atlassian Connector for IntelliJ IDEA — see [Atlassian Connector for IntelliJ IDEA](#)
- Clover IDE plugins — see the [Clover documentation](#)
- Plugins for Atlassian products — [Atlassian Plugin Exchange](#)
- Resources for plugin developers — see [Atlassian Developer Network](#)

**Atlassian IDE Connector Release Notes**

The Atlassian IDE Connectors are add-ons for your integrated development environment (IDE). They allow you to work with the Atlassian applications within your IDE. Now you don't have to switch between websites, email messages and news feeds to see what's happening to your project and your code. Instead, you can see the relevant JIRA issues, Crucible reviews and Bamboo build information right there in your development environment. You can also move quickly between your IDE and your FishEye view of your source repository.

**Atlassian Connector for IntelliJ IDEA**

**Current released version:**
Version 2.2.3 of the Atlassian Connector for IntelliJ IDEA has now been released — see the [Atlassian Connector for IntelliJ IDEA - v2.2.3 Release Notes](#).

- [Atlassian Connector for IntelliJ IDEA - v2.2.1 Release Notes](#)
- [Atlassian Connector for IntelliJ IDEA - v2.2.2 Release Notes](#)
- [Atlassian Connector for IntelliJ IDEA - v2.2.3 Release Notes](#)
- [Atlassian IntelliJ Connector 2.1.1 Release Notes](#)
- [Atlassian IntelliJ Connector 2.1 Release Notes](#)
- [Atlassian IntelliJ Connector 2.0.1 Release Notes](#)
- [Atlassian IDE Plugin 2.0 for IDEA - Release Notes](#)
- [Atlassian IDE Plugin 1.6.1 for IDEA - Release Notes](#)
- [Atlassian IDE Plugin 1.6 for IDEA - Release Notes](#)
- [Atlassian IDE Plugin 1.5 for IDEA - Release Notes](#)
- [Atlassian IDE Plugin 1.4 for IDEA - Release Notes](#)
- [Atlassian IDE Plugin 1.3.1 Release Notes](#)
- [Atlassian IDE Plugin 1.3 Release Notes](#)
- [Atlassian IDE Plugin 1.2.1 Release Notes](#)
- [Atlassian IDE Plugin 1.2 Release Notes](#)
- [Atlassian IDE Plugin 1.1 Release Notes](#)
- [Atlassian IDE Plugin 1.0 Release Notes](#)

**Atlassian Connector for Eclipse**

**Current released version:**
Version 2.0 Beta of the Atlassian Connector for Eclipse has now been released — see the release notes.

- [Atlassian Connector for Eclipse - v2.0 Beta Release Notes](#)
- [Atlassian Connector for Eclipse - v2.0 Beta Release Notes](#)
- [Atlassian Connector for Eclipse - v1.3.1 Release Notes](#)
- [Atlassian Connector for Eclipse - v1.3 Release Notes](#)
- [Atlassian Connector for Eclipse - v1.2 Release Notes](#)
- [Atlassian Connector for Eclipse - v1.1 Release Notes](#)
- [Atlassian Eclipse Connector 1.0 Beta Release Notes](#)
- [Atlassian Eclipse Connector 1.0 Beta Release Notes](#)
- [Atlassian IDE Plugin 0.4 for Eclipse - Release Notes](#)
- [Atlassian IDE Plugin 0.3 for Eclipse - Release Notes](#)
- [Atlassian IDE Plugin 0.2 for Eclipse - Release Notes](#)
- [Atlassian IDE Plugin 0.1 for Eclipse - Release Notes](#)

TreeNavigation